

NICHOLAS MCMULLIN

NUMERICAL AND EXPERIMENTAL MODELLING
OF DAM BREAK INTERACTION WITH A SEDIMENT
BED

NUMERICAL AND EXPERIMENTAL MODELLING OF DAM BREAK INTERACTION WITH A SEDIMENT BED

BY NICHOLAS MCMULLIN, MENG.

Thesis submitted to the University of Nottingham
for the degree of Doctor of Philosophy



July 2015

“The thing is rapidly approaching completion... There comes a time when one has forcibly to break off.”

— *Karl Marx [Marx, 1851]*

Dedicated to my parents.

ABSTRACT

A dam break event is considered, taking place over a uniform sediment bed. Understanding and modelling the erosion that occurs when the fluid behind the dam collapses at release has important applications in coastline morphodynamics / beach erosion modelling.

A new coupled two dimensional Navier-Stokes solver and sediment transport model is presented with novel methods for dealing with non-converging solutions to the Navier-Stokes equations and a new adaptation to the Youngs [1982] volume-of-fluid reconstruction scheme. The implementation of a sediment transport model includes a new method for accounting for mass conservation for the transition of sediment between bed and flow as well as a novel method for accounting for the redistribution of material associated with the maintenance of the critical angle of repose or slope limit. The model is validated and then applied to a dam break simulation for various backwater and tailwater conditions.

Classical experimental realizations of dam break events have involved the rapid removal of a barrier in a flume [e.g. [Levin, 1952; Dressler, 1954; Bell et al., 1992]]. However, early-time flow analysis encounters two problems with this method. Firstly, the removal of the barrier creates a strong vortex sheet on the face of the static fluid immediately behind the barrier that is not present in either the idealized problem, or the motivating environmental problems. Secondly, the removal of the barrier cannot take place instantaneously and so a brief jet-like flow is initially induced through the opening between the base of the barrier and the sediment layer. We partially circumvent these difficulties with the classical experiments by implementing a novel dam break barrier release, using a barrier similar to the wrapped-fabric design of [Dalziel, 1993], which minimizes the initial vortical disturbance.

Three-dimensional stereoscopic Particle Image Velocimetry (PIV) measurements allow us not only to capture the velocity field in the laser-plane, but also perpendicular to it. These planar experimental results are compared to the results of the numerical study and the comparison is shown to be good while the simulation successfully converges.

ACKNOWLEDGMENTS

I would like to thank everyone who has supported me throughout my university career.

Firstly, I wish to thank my supervisor, Dr. Matthew Scase, for his academic help and guidance. I would like to thank my parents and my brother for all of the help and support they have given me throughout my life. I would like to thank the various friends and housemates for their support. To my housemates: Robin Winslow, Cathal Sherry, Min-Wei Tan, Subhadra Devi Balakrishnan, Shaun Pettipiece, Joanne Pearson, Niklas Johansson and Stephen Todd. It was and is a privilege.

To my Latin and Ballroom dance partners: Antonia Mey and Joanne Pearson. Firstly, thank you for being such an important and enjoyable part of my university life. Both through dancing and through your general support. You have helped me stay somewhat sane. Toni, I blame you for leading me into this PhD, and because of that you deserve some of the credit. To all of the friends that I have made whilst dancing and competing for the University of Nottingham: Mike Hong, Ilze Skujina, Sam and Caroline Goddard, Ali King, Tasha Hall and Ricki Bills. I was never much into team sports whilst growing up, so I'm as surprised as anyone about how important being on a team with you all has been to me.

To the various international friends I have made during my time at university, particularly: Hannah Van Soelen, Maud Chenneviere, Samara Yawngghwe and Jessica Lam. I am incredibly appreciative of your friendships and thank you for being so patient when I would put off visiting due to continued PhD stress. This thesis will hopefully represent the end of that. Most of all to Hannah Van Soelen, it was to you that I often turned when things seemed at their bleakest and so it is to you that I must express my deepest gratitude.

I would also like to thank everybody who helped me to proof read this thesis: Dr. Scase, Paul McMullin, Christian McMullin, Antonia Mey, Robin Winslow, Min-Wei Tan, and Tasha Hall.

CONTENTS

List of Symbols xx

| | | |
|-------|--|-----|
| 1 | INTRODUCTION | 1 |
| 1.1 | Common terms | 1 |
| 1.2 | History of dam failure | 2 |
| 1.3 | Mathematical Study | 5 |
| 1.4 | Experimental study | 14 |
| 1.5 | Present study | 15 |
| 1.6 | Thesis Outline | 15 |
| 2 | NUMERICAL MODEL | 17 |
| 2.1 | Fluid model | 17 |
| 2.1.1 | Navier-Stokes Solution | 17 |
| 2.1.2 | Solving the Poisson equation for pressure | 22 |
| 2.1.3 | Boundary Conditions | 29 |
| 2.2 | Free Surface Tracking | 31 |
| 2.2.1 | Geometry identification | 34 |
| 2.2.2 | Volume flux calculation | 37 |
| 2.2.3 | Free Surface Boundary Conditions | 39 |
| 2.2.4 | Flotsam and Jetsam | 40 |
| 2.3 | Sediment Transport | 44 |
| 2.3.1 | Suspended Load | 44 |
| 2.3.2 | Bed Load | 48 |
| 2.3.3 | Fluid-Bed Transition | 50 |
| 2.3.4 | Recursive redistribution of suspended sediment concentration | 54 |
| 2.3.5 | Empirical Relationships | 54 |
| 2.3.6 | Critical Angle of Repose Redistribution | 57 |
| 2.3.7 | Continuity | 72 |
| 2.4 | Stability Conditions | 72 |
| 3 | NUMERICAL VALIDATION | 77 |
| 3.1 | Navier-Stokes solver efficiency investigation | 77 |
| 3.2 | Free Surface tracking tests | 87 |
| 3.2.1 | Translation | 88 |
| 3.2.2 | Rotation | 90 |
| 3.2.3 | Shear | 92 |
| 3.3 | Sediment Transport tests | 94 |
| 3.3.1 | Settling | 94 |
| 3.3.2 | Angle of repose | 95 |
| 3.3.3 | Equilibrium conditions | 96 |
| 4 | DAM BREAK CASE STUDY | 107 |
| 4.1 | Experimental apparatus and procedure | 107 |
| 4.1.1 | Image Processing | 111 |
| 4.2 | Results | 112 |

| | | |
|-------|---|-----|
| 4.2.1 | Simple vs. Wrapped-fabric release mechanism | 112 |
| 4.2.2 | Dry rigid bed | 117 |
| 4.2.3 | Tailwater shallow | 121 |
| 4.2.4 | Sediment | 123 |
| 5 | SUMMARY AND CONCLUSIONS | 127 |
| 5.1 | Summary of Thesis | 127 |
| 5.2 | Thesis Conclusions | 128 |
| 5.3 | Recommendations for future work | 129 |
| 5.3.1 | Numerical | 129 |
| 5.3.2 | Experimental | 132 |
| i | APPENDIX | 135 |
| A | SUPPLEMENTAL INFORMATION | 137 |
| A.1 | Additional free surface methods | 137 |
| A.1.1 | Marker and Cell | 137 |
| A.1.2 | SLIC | 137 |
| A.1.3 | Chorin's Method (FLAP) | 138 |
| A.2 | Initial Conditions-Circle | 139 |
| A.3 | Initial Conditions-Rotated Square | 141 |
| A.4 | User Interface | 143 |
| B | ALGORITHMS | 145 |
| B.1 | Navier-Stokes Solution | 145 |
| B.2 | Free Surface Solution | 146 |
| B.3 | Sediment Solution | 183 |
| | BIBLIOGRAPHY | 189 |

LIST OF FIGURES

| | | |
|-----------|--|----|
| Figure 1 | Teton dam failure | 5 |
| Figure 2 | Malpasset dam failure | 6 |
| Figure 3 | Shields diagram | 13 |
| Figure 4 | Cell illustration | 20 |
| Figure 5 | Red-Black Gauss-Seidel | 24 |
| Figure 6 | Multigrid V-cycle | 27 |
| Figure 7 | Grid illustration | 30 |
| Figure 8 | No slip boundary conditions | 31 |
| Figure 9 | Free slip boundary conditions | 31 |
| Figure 10 | Periodic boundary condition, horizontal | 32 |
| Figure 11 | Periodic boundary condition, vertical | 32 |
| Figure 12 | Dynamic bed boundary conditions, horizontal and vertical | 33 |
| Figure 13 | Orientations and cases for Youngs' method | 37 |
| Figure 14 | Volume for an example cell | 38 |
| Figure 15 | Vertical free surface boundary conditions | 41 |
| Figure 16 | Horizontal free surface boundary conditions | 41 |
| Figure 17 | Corner free surface boundary conditions 1 | 42 |
| Figure 18 | Corner free surface boundary conditions 2 | 42 |
| Figure 19 | Illustration of flux height | 49 |
| Figure 20 | Fluid-bed transition scenarios | 51 |
| Figure 21 | Peaks, troughs, jumps and plateaus | 59 |
| Figure 23 | P-factor and T-factor illustration | 62 |
| Figure 24 | Non-ideal interpretation of bed slopes by angle of repose redistribution algorithm | 64 |
| Figure 25 | Angle of repose redistribution area illustration | 66 |
| Figure 26 | Angle of repose redistribution, summed area illustration | 68 |
| Figure 27 | Angle of repose redistribution area illustration for trough method | 71 |
| Figure 28 | Lid driven cavity problem, streamlines | 77 |
| Figure 29 | Lid driven cavity problem, vorticity | 78 |
| Figure 30 | Method comparison, 33×33 | 79 |
| Figure 31 | Method comparison, 65×65 | 79 |
| Figure 32 | Method comparison, 129×129 | 80 |
| Figure 33 | Impact of the grid size on efficiency for the Gauss-Seidel (GS) method, where $Re=1000$. | 81 |
| Figure 34 | Impact of grid size on efficiency for the Red-Black Gaus-Seidel (RBGS) method, where $Re=1000$. | 81 |
| Figure 35 | Impact of grid size on efficiency for the Successive over-relaxation (SOR) method, where $Re=1000$. | 81 |

| | |
|-----------|--|
| Figure 36 | Impact of grid size on efficiency for the conjugate gradients (CG) method, where $Re=1000$. 82 |
| Figure 37 | Impact of grid size on efficiency for the multi-grid (MG) method, where $Re=1000$. 82 |
| Figure 38 | Impact of Reynolds number on efficiency (MG method) 83 |
| Figure 39 | Impact of grid size and Reynolds number for MG method 84 |
| Figure 40 | Impact of number of MG smoothing steps, γ_{MGV} , on efficiency 84 |
| Figure 41 | Impact of SOR factor ω on efficiency 85 |
| Figure 42 | Impact of time safety factor $\tau_{\delta t}$ on efficiency 85 |
| Figure 43 | Impact of spatial discretisation safety factor τ_{γ} on efficiency 86 |
| Figure 44 | Youngs' reconstruction example 87 |
| Figure 45 | Translation test, orthogonal 88 |
| Figure 46 | Translation test, diagonal 89 |
| Figure 47 | Rotation test 91 |
| Figure 48 | Shearing test, 500 steps 92 |
| Figure 49 | Shearing test, 1000 steps 93 |
| Figure 50 | Settling Test 94 |
| Figure 51 | Repose test, left and right plateaus 97 |
| Figure 52 | Repose test, central plateaus 98 |
| Figure 53 | Repose test, peaks and troughs 99 |
| Figure 54 | Development and equilibrium profile, uniform initial concentration 101 |
| Figure 55 | Development and equilibrium profile, initial concentration at bed 101 |
| Figure 56 | Comparison of equilibrium profiles for uniform concentration and concentration at bed initial conditions 101 |
| Figure 57 | Equilibrium results for varying sediment diameters 102 |
| Figure 58 | Equilibrium results for varying flow velocities 102 |
| Figure 59 | Equilibrium Test, variation of sediment diameter and flow velocity for constant λ 103 |
| Figure 60 | Equilibrium error plot for various ϕ factors 104 |
| Figure 61 | Equilibrium results with various ϕ factors 105 |
| Figure 62 | Barrier removal mechanisms 108 |
| Figure 63 | Experimental setup 110 |
| Figure 64 | Initial conditions 111 |
| Figure 65 | Photographic results, dry bed. $t = 0$ at initiation of barrier motion 113 |
| Figure 66 | Photographic results, dry bed. $t = 0$ at recession of barrier from fluid 113 |

| | |
|-----------|--|
| Figure 67 | Photographic results, tailwater. $t = 0$ at initiation of barrier motion 114 |
| Figure 68 | [Photographic results, tailwater. $t = 0$ at recession of barrier from fluid 114 |
| Figure 69 | Photographic results, sediment bed, tailwater. $t' = 0$ at initiation of barrier motion 115 |
| Figure 70 | [Photographic results, sediment bed, tailwater. $t = 0$ at recession of barrier from fluid 116 |
| Figure 71 | [Photographic results, sediment bed, tailwater. $t = 0$ at recession of barrier from fluid 117 |
| Figure 72 | Maximum velocity plots (PIV) for (a) a dry bed and (b) a 0.008m tailwater case, Early time case $t' = 0.02s$ 117 |
| Figure 73 | Horizontal front evolution for the dry bed case (simple release), with and without an applied offset. 118 |
| Figure 74 | Surface Profile dry rigid bed. Numerical simulation with no slip conditions. Experimentally produced surface in black, numerically produced surface in red. Surfaces shown are the produced via a contour plot describing the 0.5 boundary 119 |
| Figure 75 | Surface Profile dry rigid bed. Numerical simulation with free slip conditions. Experimentally produced surface in black, numerically produced surface in red. 120 |
| Figure 76 | Surface Profile dry rigid bed. Numerical simulation with free slip conditions and the surface condition for the F/J void filler. Experimentally produced surface in black, numerically produced surface in red. 120 |
| Figure 77 | Mean and maximum horizontal and vertical velocities for the dry bed case (simple release) 121 |
| Figure 78 | Horizontal front evolution for the non-zero tailwater case (simple release), with and without an applied offset. 122 |
| Figure 79 | Surface Profile, tailwater, free slip. Numerical simulation with free slip conditions. Experimentally produced surface in black, numerically produced surface in red. 123 |
| Figure 80 | Mean and maximum horizontal and vertical velocities for the dry bed case (simple release) 124 |
| Figure 81 | Horizontal front evolution for the non-zero tailwater over a granular bed case (simple release), with and without an applied offset. 124 |

| | | |
|-----------|---|-----|
| Figure 82 | Sediment over dry bed, surface profile, bed profile and concentration | 125 |
| Figure 83 | Sediment over dry bed, surface profile, bed profile and concentration. Separate simulation to force misinterpretation of trough. Before. $t' = 0.0861s$ | 126 |
| Figure 84 | Sediment over dry bed, surface profile, bed profile and concentration. Separate simulation to force misinterpretation of trough. After. $t' = 0.0861 + 4 \times 10^{-5}s$ | 126 |
| Figure 85 | SLIC one or two fluid cases | 138 |
| Figure 86 | Chorin's corner cases | 140 |
| Figure 87 | Areas calculation for circular initial conditions | 142 |
| Figure 88 | User interface, panel 1 | 144 |
| Figure 89 | User interface, panel 2 | 144 |

LIST OF TABLES

| | | |
|---------|---|----|
| Table 1 | Non-comprehensive sample of the most destructive dam failures in modern history | 2 |
| Table 2 | Rotation of Filled edge proportions or s-values for Youngs' method | 36 |
| Table 3 | Orthogonal Translation Errors | 89 |
| Table 4 | Diagonal Translation Errors | 90 |
| Table 5 | Rotational Errors | 91 |
| Table 6 | Shearing Errors | 93 |
| Table 7 | Repose test errors, plateaus | 96 |
| Table 8 | Repose test errors, peaks | 96 |
| Table 9 | Repose test errors, troughs | 98 |

LIST OF ALGORITHMS

| | | |
|----|--|-----|
| 1 | Case Selection for s-value calculation and flux calculation | 35 |
| 2 | | 63 |
| 3 | | 65 |
| 4 | Peak, trough, plateau and jump location algorithm . . . | 76 |
| 5 | Main Procedure | 145 |
| 6 | SLIC volume fraction Tracking method (one fluid), x-sweep | 146 |
| 7 | SLIC volume fraction Tracking method (one fluid), y-sweep | 147 |
| 8 | Chorin's FLAP volume fraction Tracking method (one fluid), x-sweep, Part 1 | 148 |
| 9 | Chorin's FLAP volume fraction Tracking method (one fluid), x-sweep, Part 2 | 149 |
| 10 | Chorin's FLAP volume fraction Tracking method (one fluid), y-sweep, Part 1 | 150 |
| 11 | Chorin's FLAP volume fraction Tracking method (one fluid), y-sweep, Part 2 | 151 |
| 12 | Young's Volume fraction Tracking method | 152 |
| 13 | Edge Intersection Calculator | 153 |
| 14 | Flux Calculator, x-sweep, Full Cells | 154 |
| 15 | Flux Calculator, y-sweep, Full Cells | 154 |
| 16 | Flux Calculator, x-sweep, V-Cells | 155 |
| 17 | Flux Calculator, y-sweep, V-Cells | 156 |
| 18 | Flux Calculator, x-sweep, H-Cells | 156 |
| 19 | Flux Calculator, y-sweep, H-Cells | 157 |
| 20 | Flux Calculator, x-sweep, BL-B or TL-B-Cells | 158 |
| 21 | Flux Calculator, x-sweep, BR-B or TR-B-Cells | 159 |
| 22 | Flux Calculator, x-sweep, BR-W or TR-W-Cells | 160 |
| 23 | Flux Calculator, x-sweep, BL-W or TL-W-Cells | 161 |
| 24 | Flux Calculator, x-sweep, BF-Cells | 162 |
| 25 | Flux Calculator, x-sweep, WF-Cells | 163 |
| 26 | Flux Calculator, y-sweep, BL-B or BR-B-Cells | 164 |
| 27 | Flux Calculator, y-sweep, TL-B or TR-B-Cells | 165 |
| 28 | Flux Calculator, y-sweep, TL-W or TR-W-Cells | 166 |
| 29 | Flux Calculator, y-sweep, BL-W or BR-W-Cells | 167 |
| 30 | Flux Calculator, y-sweep, BF or WF-Cells | 168 |
| 31 | Flux Calculator, x-sweep, A-Cells, Part 1 | 168 |
| 32 | Flux Calculator, x-sweep, A-Cells, Part 2 | 169 |
| 33 | Flux Calculator, y-sweep, A-Cells, Part 1 | 170 |
| 34 | Flux Calculator, y-sweep, A-Cells, Part 2 | 171 |

| | | |
|----|---|-----|
| 35 | Flux Calculator, x-sweep, B-Cells, Part 1 | 172 |
| 36 | Flux Calculator, x-sweep, B-Cells, Part 2 | 173 |
| 37 | Flux Calculator, y-sweep, B-Cells, Part 1 | 174 |
| 38 | Flux Calculator, y-sweep, B-Cells, Part 2 | 175 |
| 39 | Flux Calculator, x-sweep, C-Cells, Part 1 | 176 |
| 40 | Flux Calculator, x-sweep, C-Cells, Part 2 | 177 |
| 41 | Flux Calculator, y-sweep, C-Cells, Part 1 | 178 |
| 42 | Flux Calculator, y-sweep, C-Cells, Part 2 | 179 |
| 43 | Flux Calculator, x-sweep, D-Cells, Part 1 | 180 |
| 44 | Flux Calculator, x-sweep, D-Cells, Part 2 | 181 |
| 45 | Flux Calculator, y-sweep, D-Cells, Part 1 | 182 |
| 46 | Flux Calculator, y-sweep, D-Cells, Part 2 | 184 |
| 47 | Peak, Trough, Jump and Plateau Finder | 185 |
| 48 | Angle of repose height redistribution | 186 |
| 49 | Sediment Procedure, part 1 | 187 |
| 50 | Sediment Procedure, part 2 | 188 |

ACRONYMS

| | |
|------|------------------------------------|
| CG | conjugate gradients |
| FLAP | Flame Advection and Propagation |
| GS | Gauss-Seidel |
| MG | multigrid |
| NS | Navier-Stokes |
| ND | non-dimensional |
| NSWE | Nonlinear Shallow Water Equations |
| PIV | Particle image velocimetry |
| PPE | Pressure Poisson Equation |
| RBGS | Red-Black Gauss-Seidel |
| SMAC | Simple Marker and Cell |
| SLIC | Simple Linear Interface Calculator |
| SOR | Successive over-relaxation |
| YVOF | Youngs' Volume of Fluid |

LIST OF SYMBOLS

| | | |
|---------------|--|----|
| τ_* | Shields Parameter [] | 12 |
| τ' | Bed shear stress [M/LT ²] | 12 |
| ρ'_s | Sediment density [M/L ³] | 12 |
| ρ' | Fluid density [M/L ³] | 12 |
| g' | Acceleration due to gravity [L/T ²] | 12 |
| d' | Characteristic particle diameter [L] | 12 |
| u' | Horizontal velocity component [L/T] | 18 |
| v' | Vertical velocity component [L/T] | 18 |
| t' | Time [T] | 18 |
| x' | Horizontal length [L] | 18 |
| y' | Vertical length [L] | 18 |
| p' | Pressure [M/LT ²] | 18 |
| μ' | Dynamic Viscosity [M/LT] | 18 |
| g'_x | Horizontal body acceleration [L/T ²] | 18 |
| g'_y | Vertical body acceleration [L/T ²] | 18 |
| u | non-dimensional (ND) horizontal velocity component [] | 18 |
| v | ND vertical velocity component [] | 18 |
| t | ND time [] | 18 |
| x | ND horizontal length [] | 18 |
| y | ND vertical length [] | 18 |
| p | ND pressure [] | 18 |
| g_x | ND horizontal body acceleration [] | 18 |
| g_y | ND vertical body acceleration [] | 18 |
| U' | Velocity scale [L/T] | 18 |
| L' | Velocity scale [L/T] | 18 |
| P' | Pressure scale [M/LT ²] | 18 |
| Re | ND Reynolds number [] | 18 |
| n | Timestep index [] | 19 |
| δt | ND Timestep size [] | 19 |
| F | Prediction for u [] | 19 |
| G | Prediction for v [] | 19 |
| u_{max} | Maximum value of u [] | 22 |
| v_{max} | Maximum value of v [] | 22 |
| i | Horizontal cell index [] | 21 |
| j | Vertical cell index [] | 21 |
| δx | ND cell width [] | 21 |
| δy | ND cell height [] | 21 |
| γ | Donor cell weighting factor [] | 21 |
| τ_γ | Gamma safety factor | 22 |
| r^{it} | Residual error for current iteration, it [] | 23 |
| rhs | Right hand side of the Poisson equation for the pressure [] | 23 |

| | | |
|-----------------------|--|----|
| $\ r^{it}\ _2$ | L^2 -norm for the residual error [] | 23 |
| i_{\max} | Total number of real columns [] | 23 |
| j_{\max} | Total number of real rows [] | 23 |
| n_{cols} | Total number of columns, including ghost columns [] | 23 |
| n_{rows} | Total number of rows, including ghost rows [] | 23 |
| ω | Relaxation factor for successive over relaxation | 25 |
| φ | Conjugate vector | 26 |
| $r2\text{sum}$ | Sum of residual error squared | 26 |
| φsum | Sum representing $\phi^T \mathbf{A} \phi$ | 26 |
| η | Factor representing the change in pressure/residual error | 26 |
| ξ | Factor applied as the change in the conjugate vector | 26 |
| \mathcal{R} | Restriction factor | 27 |
| \mathcal{P} | Prolongation factor | 27 |
| t_c | Time taken to compute | 29 |
| r_b | ND bed height within the highest bed cell [] | 33 |
| c | Volume Fraction [] | 33 |
| n^x, n^y | Angle calculation stencil [] | 34 |
| β | Interface angle within a given cell [°] | 34 |
| α | Square cell equivalent interface angle [°] | 34 |
| s_L | Proportion of cell left hand edge in contact with fluid | 35 |
| s_R | Proportion of cell right hand edge in contact with fluid | 35 |
| s_T | Proportion of cell top edge in contact with fluid | 35 |
| s_B | Proportion of cell bottom edge in contact with fluid | 35 |
| Fl | ND outward volume flux through left cell edge [] | 37 |
| Fr | ND outward volume flux through right cell edge [] | 37 |
| Ft | ND outward volume flux through top cell edge [] | 37 |
| Fb | ND outward volume flux through bottom cell edge [] | 37 |
| \mathbf{v} | Vector form of velocity [] | 38 |
| c^* | Intermediary volume fraction [] | 38 |
| n_x, n_y | x and y components of the outward unit normal vector [] | 40 |
| m_x, m_y | x and y components of the tangential vector [] | 40 |
| C' | Sediment concentration [M/L ³] | 45 |
| K'_x | Longitudinal coefficient of Diffusion [1/LT] | 45 |
| K'_y | Vertical coefficient of Diffusion [1/LT] | 45 |
| C | ND sediment Concentration [] | 45 |
| ρ'_{pack} | Packed bed density/concentration [M/L ³] | 45 |
| K_x | ND horizontal diffusion coefficient [] | 45 |
| K_y | ND vertical diffusion coefficient [] | 45 |
| v'_s | Settling velocity [L/T] | 45 |
| v_s | ND settling velocity [] | 46 |
| h' | Bed height [L] | 48 |
| q'_s | Bed load [M/T] | 48 |
| $q'_{s,v}$ | Bed load [L ³ /T] | 48 |
| Q | ND concentration deposition rate [] | 48 |
| S | ND concentration erosion rate [] | 48 |
| τ_D | Critical stress threshold for deposition [] | 48 |

| | | |
|--------------------|---|-----|
| u_* | ND bed shear velocity [] | 49 |
| q_s | ND bed load [] | 49 |
| h | ND bed height [] | 49 |
| r | ND fluid remainder height within the highest bed cell [] | 49 |
| FH | ND sediment flux height [] | 49 |
| H^* | ND height change due to deposition or erosion [] | 49 |
| h_{qs} | ND height change due to bed load [] | 50 |
| κ | von Karman constant, usually quoted as 0.4 [] | 55 |
| z_0 | Friction length [] | 55 |
| C_A | Empirical factor [] | 56 |
| C_B | Empirical factor [] | 56 |
| ϕ | Fluid damping coefficient [] | 56 |
| ϵ_f | Coefficient of fluid momentum/kinematic eddy viscosity [] | 56 |
| $\epsilon_{f,max}$ | Maximum Coefficient of fluid momentum/kinematic eddy viscosity [] | 56 |
| d_f | ND total fluid depth at a given width [] | 56 |
| F_P | P-factor or “peakiness” quotient [] | 59 |
| F_T | T-factor or “troughiness” quotient [] | 60 |
| $sum_{L,p}$ | Sum of sediment heights left of a peak [] | 61 |
| $sum_{R,p}$ | Sum of sediment heights right of a peak [] | 61 |
| $sum_{L,t}$ | Sum of sediment heights left of a trough [] | 61 |
| $sum_{R,t}$ | Sum of sediment heights right of a trough [] | 61 |
| PTL | Peak/trough locations [] | 61 |
| α | Index for current peak/trough location [] | 61 |
| θ_c | Critical angle of repose [rad] | 65 |
| B | ND critical angle of repose triangle base [] | 65 |
| H | ND critical angle of repose triangle height [] | 65 |
| A | ND critical angle of repose triangle area [] | 65 |
| Z_1 | Even/odd modifier for ascending slopes (peak) [] | 67 |
| Z_2 | Even/odd modifier for descending slopes (peak) [] | 67 |
| PL | Plateau length [] | 67 |
| B_i | Equivalent column width of B [] | 69 |
| A_E | Extra area due to smoothing beyond domain [] | 69 |
| H_E | Height equivalent of Extra area A_E [] | 69 |
| H_E | Height equivalent of Extra area A_E [] | 69 |
| γ_{MGV} | Number of smooth iterations per level for the multigrid V-cycle [] | 83 |
| I_L | Boolean index showing existence of fluid in left hand cell | 138 |
| I_R | Boolean index showing existence of fluid in right hand cell | 138 |

INTRODUCTION

Dam break dynamics is the area of study concerned with the morphodynamics of a collapsing block of fluid that had previously been retained. Dam break is both a classical problem and a continuing real world problem, the effects of which include injury, loss of life and destruction of property and infrastructure. The key parameters of a dam break are the reservoir depth and volume. The key flow properties that result from a failure are the fluid depth and velocity. These initial parameters and resultant properties are used to form predictions about the impact of the flow. The impact of such a flood is also dependant on the interaction of the fluid with its surroundings, specifically the potentially erodible bed. A dam break wave can erode the bed and move the eroded materials downstream, compounding both the lethality and the destructive potential of the flow [Capart and Young, 1998].

This introduction will discuss the history of dam failure in order to give context and motivation for the current work. This is followed by a brief summary of academic study to date in the field of dam break flow and an outline for the work that was undertaken as part of this research project.

1.1 COMMON TERMS

A dam is a barrier across a waterway that controls the flow or raises the water level, thus we can define dam failure as any occurrence where water passes the barrier in an unintended and uncontrolled way. Dam break is the name given to describe the idealised release of a reservoir of water akin to that of a complete collapse of a dam. For simplicity, this thesis will use the term 'dam failure' only to refer to real life occurrences and 'dam break' to refer to the idealised classical open channel problem, a rectangular reservoir with a finite backwater length, held by a negligibly thin barrier (also known as a dam, gate or lock) that is instantaneously removed allowing the liquid of the reservoir to flow forward over a horizontal bed. The resultant dam break flow is a type of 'gravity current'. A gravity current is a flow driven by the density difference between the fluids involved. In the case of a dam break flow, the liquid in the reservoir is of high density compared to the atmosphere ahead of it and so the liquid will flow forward and underneath the atmosphere, whilst the atmosphere will flow backward and over the top of the liquid. The high density difference causes a high velocity flow.

Other important terms used to describe the geometry of the scenarios and the flow include the backwater, which is another name for the reservoir, and the tailwater, which is the name given to any liquid downstream of the dam. The part of the dam break wave that extends the furthest forward is known as the wave front, or simply the front. In the case that a gravity current front disconnects from the surface it is running over, it is said to have a nose (the part of the wave furthest forward) and a foot or toe (the most forward part of the wave still in contact with the floor surface).

1.2 HISTORY OF DAM FAILURE

The construction of dams is not a solely modern practice. The first known dam was constructed approximately 5000 years ago in Jawa, Jordan [Fahlbusch, 2009], and dam failure at varying scales is likely to have occurred over the entire period since then. The modes by which a dam can fail are by overtopping, where water passes over the crest of the dam, or breach, where water passes through the dam. There are numerous causes of dam failure including heavy rainfall, snowmelt, inadequate spillway capacity, internal erosion, poor construction or maintenance, poor foundation, geological instability, earthquake and through destructive acts. Table 1 shows a small sample of some of the most destructive dam failures that have occurred in recent history.

| Year | Dam name | Location | Fatalities | Damages |
|------|------------------|-----------|------------|---------------------------------|
| 1889 | South Fork | PA, USA | 2209 | \$17 million ¹ |
| 1917 | Tigra | India | 1000 | Unknown ² |
| 1928 | St. Francis | CA, USA | > 600 | > \$5.5 million ¹ |
| 1959 | Malpasset | France | 421 | Unknown ³ |
| 1963 | Vajont | Italy | 2600 | Unknown ^{3,4} |
| 1967 | Sempor | Indonesia | > 2000 | Unknown ⁴ |
| 1972 | Buffalo Creek | WV, USA | 125 | \$400 million ¹ |
| 1972 | Canyon lake | SD, USA | 237 | \$60 – 164 million ¹ |
| 1975 | Banqiao et alibi | China | 171000 | Unknown ⁵ |
| 1976 | Teton | ID, USA | 11 | > \$1 billion ¹ |
| 1979 | Machhu II | India | > 1000 | Unknown ² |

Table 1: Non-comprehensive sample of the most destructive dam failures in modern history. Note that the number of fatalities quoted for the various Chinese dam collapses of 1975 ranges between 26000 (direct) and 230000 (indirect) and for the Machhu-II dam failure between 1000 and 25000. ¹ ASDSO [2013], ² Jain et al. [2007], ³ Shaffner [2011], ⁴ Wohl [2013], ⁵ Graham [1999]

The Teton dam failure in 1976 is a highly cited example of dam failure, for which a lot of attention was devoted. It was one of the largest and most destructively costly dam failures to occur in the United States of America and one of very few dam failures to be captured on film as it happened, see figure 1. The Teton dam failed when water internally eroded through the earthen dam by a process called ‘piping’ [Smalley and Dijkstra, 1991], causing the structure to initially leak and then completely fail. The worsening of the leak occurred over 4 hours, but, when it fully breached, the force of water caused rapid collapse of the dam wall, unleashing the remaining quantity of the water held by the dam [Chadwick and Casagrande, 1976]. The failure caused the death of 11 people and the destruction of thousands of properties. The force of the water and the deposition of material also destroyed the banks of the river and its ecology. The Teton dam failure mirrored an incident that occurred at the Fontenelle dam in 1965, approximately ten years prior to the Teton dam failure, where a dam collapse was evaded through a lowering of the reservoir through the outlet works when leakage due to erosional piping was discovered. If the problems of Fontenelle had been properly understood and disclosed, it is thought that the failure at Teton could have been avoided [Shaffner, 2011].

The Malpasset Dam failure of 1959 is an example of catastrophic breach. The dam failure has been attributed to previously unknown geological instability (tectonic fault) in conjunction with high rainfall. The catastrophic breach released $49 \times 10^6 \text{m}^3$ of water which reached speeds of 20ms^{-1} . The flooding caused up to 10m depth of erosion near the dam site and up to 5m depth of erosion in the city of Fréjus, causing significant urban destruction [Bellaiche, 1969; Mulder et al., 2009]. The scale of the dam failure is illustrated by the ruined remains of the dam at Malpasset, see figure 2.

The failures at Teton and Fontenelle were, in part, due to the design and construction methods, however many historical dam failures have been as a result of highly improbable events. This can come in the form of a severe storm, which was the case of the Banquiao dam (and the surrounding dams that also broke) during Typhoon Nina in 1975, a destructive act, which was the case for various dam breaks, including Möhne, during the dam busters campaign inflicted by the Royal Air Force during World War II, or by landslide which was the case of the Vajont dam failure in 1963. A landslide into the reservoir at Vajont displaced the reservoir water, causing the entire reservoir contents (approximately $50 \times 10^6 \text{m}^3$ of water) to overtop the dam, which remained standing, relatively undamaged [Shaffner, 2011]. This resulted in a wave that was approximately 250m high which ran downstream destroying 5 villages and killing 2600 people, leaving the entire basin under a thick layer of mud and debris. The design of the dam structure was sufficient up until that point, however the geologi-

cal instability of the surrounding mountains was underestimated and as such a landslide was not planned for.

Although most of the largest and most destructive dam failures on record occurred in the 20th Century, dam failures continue to occur. The recent Ivanovo Dam failure in 2012 was an example of heavy snowmelt and poor maintenance (cracking went unrepaired) leading to a collapse that caused flooding, destruction of property in two villages and the death of 8 people [Su et al., 2013].

The study of these failures by geologists and civil engineers is important so as to understand the modes by which failure is possible, design for them, and thus reduce the frequency of failure occurrence. Dams are designed such that they can control input volume flow rates equivalent to a given storm or flood probability, for example a one in one thousand year flood. It is possible, however, for rainfall to exceed this capacity in a rarer flood or storm event, causing a failure of the dam. The physical and economic limits of construction make it impossible to design for all cases and as such failure can never be prevented entirely. It may, however, be possible to mitigate the effects of the failure through the study of the resulting flows. Mitigation efforts may come in many forms, including consideration of other locations, relocation of at-risk residents, creation of diversionary channels or protective structures and the development of warning systems. In each of these cases, a greater understanding of the flows that could potentially be generated in a failure can help properly design the mitigative strategies.

Graham [1999] states that the numbers of lives lost as a result of a dam failure is most dependant on the number of people occupying the flood plain, the amount of warning that is provided to them and the scale of the resultant flood. Improved simulation of these flows can allow for better understanding of the scale of a potential flood and, as such, a more accurate calculation of risk to human life can be made. Through this, it may be possible to develop better risk reduction strategies for currently standing dams and dams that are planned to be built.

In addition to its direct application to dams, the study of dam break flow has important applications in coastal morphodynamics, including levee overtopping (e.g. Hurricane Katrina, 2005) and Tsunami run-up (e.g. Indian Ocean Tsunami, 2004). The proper understanding of how a dam break flow interacts with erodible material and sediment is also of particular interest in the case of deliberate fluid release for the purpose of moving or redistributing sediment. This action was taken in Glen canyon, where in a restorative effort, a man-made flash flood was created by opening the Glen canyon dam outlets to mimic the yearly floods (that no longer occur after the construction of the dam) that would clear the river basin of encroaching vegetation and redistribute soil and rocks deposited by rock slides from the canyon

walls [US Department of the Interior, 1996]. Another example of a similar application is in the use of large releases of water to remove solid deposits in sewers. In these cases the interaction between the flow and the sediments are of key importance. Improved modelling of flow-sediment interaction is of use for various engineering applications, and as such the focus of this thesis will be in this area.



Figure 1: The teton dam shortly after breach

1.3 MATHEMATICAL STUDY

The dam break problem can be mathematically modelled using the depth averaged Nonlinear Shallow Water Equations (NSWE), which describe the flow beneath a surface and are derived from the conservation of momentum and conservation of mass equations that are known as the Navier-Stokes (NS) equations. The one dimensional form of the NSWE, also known as the Saint-Venant equations [Saint Venant, 1871], were originally analytically solved for the dam break problem by Ritter in 1892. Using the method of characteristics, Ritter developed an analytical solution of the one dimensional NSWE for a dam break with an infinite backwater length, zero tailwater height and smooth bed. This resulted in equations for the flow profile and velocity and results for the positive and negative wave velocities (or celerities) and the constant flow height at the dam location. Stoker [1957] went on to produce an analytical solution for dam break with a non-zero tailwater height. This result produces a wave that moves through the backwater causing a recession of backwater height and a wave that moves through and over the tailwater causing an increase in flow depth downstream.

These solutions failed to account for the impact of bed resistance for which a few approaches were developed. Dressler's approach (firstly proposed in 1952 and then compared to experimental results



Figure 2: The remains of the Malpasset dam

in 1954) introduced a resistance term to the NSWE. Whitham [1955] modelled resistance with the assumption that it only existed in the tip, in a region near to the wave front. Behind this region Whitham's solution uses the Ritter solution for no bed resistance. The Karman-Pohlhausen method for the solution of the boundary layer equations was applied to a boundary layer at the tip. The results produced good agreement with those of Dressler's model. Hogg and Pritchard [2004] used a third approach, also with a separation of a tip region, producing similar but, not identical, results. The current model does not account for hydraulic resistance beyond the slip conditions given at the lower edge of the simulation. Dressler posited that it is an appropriate approximation for small time spans, as the fluid, accelerating from rest, will not have had sufficient time to develop resistive action with turbulence [Dressler, 1954].

The analytical solutions produced by Ritter, Dressler, Whitham and Hogg and Pritchard are only valid when the flow has moved some distance and the velocity of the flow is predominantly horizontal. The initial stages of the flow were analysed by Stoker [1957] using a Lagrangian approach based on the work of Pohle [1950], and by Korobkin and Yilmaz [2009] using an Eulerian approach based on the work of King and Needham [1994]. All of these methods failed to properly recreate the surface profile at the tip.

Recent analytical studies have been focussed on analysing the flows where there is a discontinuity of bed level or a step. The dam break problem for non-zero tailwater conditions was analytically solved by [Bernetti et al., 2008], which showed a stationary shock at the location of the bed step and the possible resulting wave structures.

Numerical modelling of Dam break interaction with movable beds has also been of particular interest in recent years. Capart and Young [1998] began study on dam break interactions with erodible beds through numerical solution of the NSWE and experimental study. Their study found that the dam break wave caused bed sediment to be lifted and suspended high into the wave and that the interaction between the sediment and wave itself caused the wave to break backwards, forming a hydraulic jump. Fraccarollo and Capart [2002] continued the work in this field developing an analytical solution and providing a new set of experimental results, which showed good agreement with the results of Capart and Young [1998] but differed in the existence of the hydraulic jump. The main difference between these experiments was the material that was used. In the experiments presented by Capart and Young [1998], the sediment was a light spherical artificial pearl of density hardly greater than that of the water ($\rho_s = 1048 \text{ kg m}^{-3}$). The second set of experiments compared to by Fraccarollo and Capart [2002], performed at Louvain-la-Neuve (Université Catholique de Louvain), used a denser sediment, made up of PVC pellets ($\rho_s = 1540 \text{ kg m}^{-3}$). Kelly and Dodd [2009] also

modelled dam break over a dry movable bed which resulted in a similar flow pattern to that produced by Fraccarollo and Capart [2002], with differences in the tip region. In a review of analytical work on the dam break problem, Zhu [2012] noted this was probably due to different assumptions made about the vertical structure at the tip.

Cao et al. [2004] presented a model for the simulation of dam break over mobile beds, solving the 1D Shallow Water Equations using a weighted average flux scheme in conjunction with the HLLC approximate Riemann solver and SUPERBEE limiter [Toro, 2001]. This model also captures a hydraulic jump, formed at the dam site due to rapid bed erosion. The hydraulic jump is found to attenuate as it propagates upstream, until it disappears. Backward wave speed or celerity is found to match that of dam break over a fixed bed, however the forward celerity is greatly reduced as a result of the bed erosion (also causing greatly modified free surface profiles and hydrographs). The results showed that the mobile bed could undergo significant scour, with erosional depths of the same magnitude as the flow depth, and that there was active sediment exchange between the water column and the bed (with a sharp spatial gradient of concentration). These features were seen to be highly dependant on sediment particle size, with finer sediment particles causing greater bed mobility. The model showed good agreement with the results of Fraccarollo and Capart [2002] and the experiments of Capart and Young [1998] in the intermediate range. Wu and Wang [2007] also presented results of a 1D generalised Shallow Water Equation model for the simulation of dam break over a movable bed, with a nonequilibrium sediment transport model and the van rijn approach for bed load. The model is solved using the Finite Volume Method with a first order upwind scheme. The model was found to have sensitivity to the sediment nonequilibrium adaptation length, the Manning's roughness coefficient and a proposed correction factor.

Zech et al. [2008] studied both near and far field responses to a dam break wave through the implementation of a two layer scheme which attempts to separately model clear (sediment free) and turbid (sediment containing) water. The scheme differs from the Method presented by Capart [2000] on which it is based, by allowing the layers to have distinct velocities and concentrations. The model was validated against data from experiments on the flat [Spinewine, 2005] and stepped bed cases [Spinewine and Zech, 2003]. This study particularly noted the impact that sediment mobilization had on the wave structure, delaying the front but increasing the water depth behind the front.

Xia et al. [2010] presented a numerical solution to an adapted form of the two dimensional Shallow Water Equations for turbid water, coupled with nonequilibrium sediment transport equations for graded sediment and a bed evolution model. The model was numerically

solved via an unstructured Finite Volume Method with an approximate Riemann solver (based on Roe-MUSCL [Roe, 1981]). The model was validated against previous experimental models and experimental data Spinewine and Zech [2007]; Zech et al. [2008] and it was found that, for the initial stages, the rate of bed evolution to be comparable to the rate of water depth change. It was also found that overall erosion was greater for uniform beds than non-uniform beds, however the maximum erosion depth was greater for uniform beds.

El Kadi Abderrezzak and Paquier [2010] investigated ten different sediment transport formulas, applied to a 1D, One Layer model using the Shallow water equation, Exner equation, with nonequilibrium sediment transport. El Kadi Abderrezzak and Paquier found that formulas proposed by Meyer-Peter and Müller [1948] (with a factor adaptation), Smart and Jaggi, Cheng [2002], Abrahams [2003] and Camenen and Larson [2005] ranked highest in predictive capability when compared to the experimental results of Spinewine [2005]; Spinewine and Zech [2007] for a range of flow conditions. He also found the impact of the critical shields number and the method by which bed failure or avalanching is imposed to be non-negligible.

Zhang and Duan [2011] presented a 1D Finite Volume model for unsteady flows such as dam break over a mobile bed. It comprised a modified 1D Shallow Water Equation, a nonequilibrium/Exner equation hybrid sediment transport model and the Van Rijn [1984a,b] approach for bed load. Solutions were computed using the Finite Volume Method and compared four different flux schemes: upwind flux (see Capart and Young [1998]; Fraccarollo and Capart [2002]; Wu and Wang [2007], weighted average flux (see Cao et al. [2004]), HLL and HLLC [Harten et al., 1983; Toro et al., 1994] and Roe's scheme [Roe, 1981]. Reasonable agreement was found however the model failed to capture the hydraulic jump. Results produced using the weighted average flux scheme produced some surface and bed profile oscillations and over predicted scour.

Zhang et al. [2014] presented a coupled 1D/2D hybrid model of dam break over mobile beds, solving the generalised Shallow Water Equations, nonequilibrium sediment transport and bed change (with variable flow density). The scheme implements an explicit Finite Volume Method with the HLL approximate Riemann solver, on an unstructured, nonstaggered (collocated) quadtree rectangular grid [Zhang and Wu, 2011]. 2D models suffer from increased complexity, cost and data requirements and as such this hybrid method attempted to approach the increased simulation performance of 2D models with reduced computational effort requirements.

Recently some alternative approaches have been presented to the solution of this problem. Marsooli and Wu [2014] approached the problem of dam break over a movable bed through the solution of the 3D Reynolds Averaged Navier-Stokes equations with nonequilib-

rium sediment transport equations. The scheme implemented a Finite Volume Method on a collocated hexahedral mesh, using the Volume of Fluid (VOF) method for evolution of the free surface and the moving mesh technique for bed evolution. The results were compared to previous 1D and 2D models and found significant improvement particularly in the initial stages. Like [Wu and Wang \[2007\]](#), Marsooli's model was found to be sensitive to bed friction and adaptation length parameters.

[Razavitoosi et al. \[2014\]](#) used a Lagrangian approach to the Navier-Stokes equations (in 2D) by using the smoothed particle hydrodynamics method (a method originally developed for astrophysical problems [[Lucy, 1977](#); [Gingold and Monaghan, 1977](#)]). Fluid and sediment phases are described by particles of weakly compressible fluids. Incompressibility was achieved using a state equation that permits a maximum of a one percent change in density fluctuation. Water phase is modelled as Newtonian but the sediment phase is modelled as a non-Newtonian fluid, and three different rheological models were tested, 'artificial viscosity' [[Monaghan, 1994](#)], 'Bingham and artificial viscosity' [[Capone, 2009](#)] and 'Bingham and Cross' [[Shao and Lo, 2003](#)]. The lowest error was obtained through the use of the combined Bingham model with artificial viscosity. Free surface profiles show good agreement however predicted sediment height had an average relative difference of approximately 9%.

[Kesserwani et al. \[2014\]](#) presented a 1D model of dam break flows over a movable bed with sediment transport. The model sought to solve the Shallow Water Equations using a second order, Runge-Kutta, discontinuous galerkin model, with consideration for sediment transport and bed evolution. Reasonable agreement was found to the Taipei [[Capart and Young, 1998](#)] and Louvain [[Fraccarollo and Capart, 2002](#)] experimental data sets, however, it was found that its applicability was highly dependant on calibration of sediment parameters or each specific test configuration.

In a recent experimental paper by [Soares-Frazão et al. \[2012\]](#), the experimental initial conditions of a dam break over a mobile bed were given to various individuals and teams from various institutions, who were asked to model the flow height distribution and final bed condition. These teams were not given access to the experimental results and as such, the comparison of the results serves as a benchmarking review of many of the current numerical models. The models predominantly used the Shallow Water Equations or the Two Layer approach [[Capart, 2000](#); [Spinewine, 2005](#); [Swartenbroekx et al., 2010](#)] as governing equations with Exner, Advection or nonequilibrium equations for sediment transport. These were predominantly solved using a Finite Volume or Finite Element scheme. The models all tended to accurately predict the evolution of the flow profile and the location of scour near the dam site, however they all tended to underestimate

the amplitude of scour and deposition and did not accurately place the shape and location of deposition. The governing equations and numerical solution methods used were found to have less of an impact than the choice of sediment closure equations (for example the use of the Meyer-Peter and Muller 1948 formula for bed load and the Shields 1936 Diagram for critical bed stress).

The focus of the current work was to attempt to create a model that coupled a fluid and a sediment transport model that could be applied to many scenarios without neglecting the impact of vertical velocities. The present model is a numerical solution of the two dimensional NS equations for laminar flow, through the Finite Difference Method, coupled with a free surface and sediment evolution model. This is a departure from the methods used by Capart and Young, Fraccarollo and Capart and many of the other authors detailed above, which are based on the assumption that vertical accelerations and vertical velocity distribution can be neglected, which is not the case for the initial stages of a dam break flow. The initial stages of a dam break flow will have large vertical velocities downwards and it is hypothesized that these velocities will be a primary cause of sediment erosion and suspension into the flow. A similar approach has recently been independently and concurrently taken by Burkow and Griebel [2013], however, the present study has generated various computational and experimental methods and aims to provide experimental comparison.

When solving the NS equations for a fluid with a free surface, one must calculate, and track, the position of the free surface separately to the calculation of the pressure and velocity fields via the solution of the Navier-Stokes equations. Various methods have been used to do this, which can generally be split into the categories of surface tracking, movable mesh and volume tracking methods [Hyman, 1984]. Surface tracking methods define and track points on the surface of the fluid, for example the Level Set method. Movable mesh methods deform the mesh to keep the interface on cell edges and diagonals, instead of having a rigid mesh of cells with the interface potentially crossing the cell at any point. Volume tracking methods track the location of the fluid as a whole. These methods can generally be further subdivided into Marker and Cell methods (or particle in cell methods) which define and follow virtual particles placed in the fluid, and volume fraction tracking methods, which follow the fluid by defining a volume fraction in each cell where 0 is an empty cell, 1 is a full cell and a number between 0 and 1 contains an interface. The volume fraction tracking methods are well suited to models using advection-diffusion equations (which was the intention for the solution of suspended sediment transport) and thus the present study implements a volume fraction tracking method.

Initial volume fraction tracking methods approximate the geometry of the cell as a rectangle (SOLA [Hirt et al., 1975], SLIC [Noh

and Woodward, 1976], SOLA-VOF [Nichols et al., 1980], VOF [Hirt and Nichols, 1981]). The stair-stepped methods added the geometries of rectangles with rectangular corner voids (Flame advection and propagation method, Chorin [1980]). Improvements in accuracy were gained by approximating the curved interface within a cell as an angled, straight line (PLIC (see Kothe and Rider [1995a,b]), Youngs' method [Youngs, 1982]). An adaptation of Youngs' method is used in the present model, due to its balance of accuracy and low computational cost [Rudman, 1997].

In order to calculate the impact of the flow on the sediment bed, a sediment transport model must be included. Sediment transport has classically been split into suspended sediment load and bed load, where suspended sediment is carried by the bulk motion of the flow and bed load includes motion of sediment along the bed. The definition of what constitutes bed load and what constitutes suspended load varies between academic studies, however bed load is most commonly defined as either motion by the modes of rolling, sliding and saltations or small jumps [Bagnold, 1966], or as motion within a layer two particle diameters thick [Einstein, 1950]. Suspended load is thus all other sediment transport in the bulk of the flow. Motion of the sediment occurs when the drag and lift forces overcome the submerged weight and frictional forces of a given sediment particle. The Shields parameter τ_* represents the balance of these forces and is defined as

$$\tau_* = \frac{\tau'}{(\rho'_s - \rho')g'd'} \quad (1)$$

where τ' is the bed shear stress, ρ'_s is the sediment density, ρ' is the density of the liquid, g' is the acceleration due to gravity and d' is the characteristic particle diameter, often chosen to be the mean sediment grain diameter.

Initially the threshold for the initiation of sediment motion was studied for unidirectional, steady state flows like those found in rivers. Hjølstrom [1935, 1939] and Sundborg [1956] present graphs of the average river flow velocity which will successfully transport a sediment particle of a given size. Laboratory experimentation into steady state open channel flow conducted by Shields [1936] (and later by Bagnold and Taylor [1963]) resulted in a means to calculate a threshold parameter, the critical Shields number θ_c , which would indicate the initiation of motion. These results are commonly referred to as the Shields diagram. An approximation to the Shields diagram is given by Van Rijn [1984a] for direct calculation of the critical Shields parameter for a given dimensionless particle diameter, see figure 3.

In the near bed region, the lift imparted to a sediment particle is caused by the local turbulence experienced. This turbulence has been shown to scale well with a local shear velocity parameter u_* for steady and uniform cases, however this has not been shown to be reliable in unsteady cases where variations in bed turbulence can vary

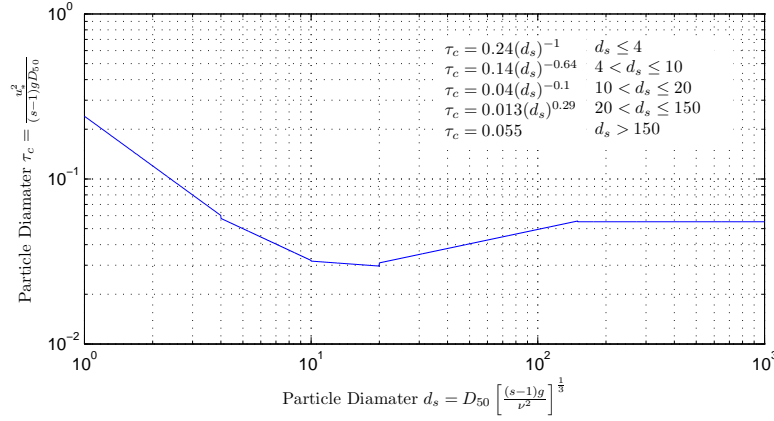


Figure 3: Approximation to Shields diagram from the equations presented by Van Rijn [1984a]

greatly [Shvidchenko and Pender, 2000; Wu et al., 2008]. Komar and Miller [1973] looked at the threshold for sediment motion beneath oscillatory water waves using the data produced by Bagnold and Taylor [1946] and Manohar [1955]. Komar and Miller notes that the stress (and thus the probability that sediment motion initiates) is greater in an accelerating flow than in a steady unidirectional flow with an identical instantaneous velocity. Despite this, the Shields parameter continues to be the most commonly used criterion for the initiation of sediment motion at the bed. The Shields parameter is used as an approximation for the testing of the new geometric conservation of sediment mass conditions at the bed fluid transition, with a view to the future inclusion of a turbulence model. The evolution of the bed is most commonly modelled via the Exner equation [Exner, 1925] which is a statement of conservation of mass in terms of bed height change. This statement requires the separate calculation of the bed load. Various empirical studies have determined different empirical relationships for the bed load, however, the relationships presented by Einstein [1950] and Meyer-Peter and Müller [1948] are the most commonly used. The present work uses the Meyer-Peter and Müller relationship, following the work of Burkow and Griebel [2013].

The transport of suspended sediment is governed by the conservation of sediment mass, which reduces to an advection-diffusion equation for sediment concentration [Lesser et al., 2004]. At the bed surface, the upward motion of the particles is only due to local turbulence, however, in the main bulk of the flow, the lift forces are imparted by vertical bulk motions of the liquid (accounted for by the advection terms) as well as local turbulence (accounted for by the diffusion terms). In addition to these terms, the advection-diffusion equation has terms to account for sediment weight through a settling velocity term, and transition to and from the bed through a sink/-source term. The application of these equations allows a simulation

to calculate the impact of the flow velocity on the bed heights and the sediment concentrations through the flow. These changes then impact the calculation of the pressure and velocities within the flow through changes in bed boundary conditions and changes in flow domain.

1.4 EXPERIMENTAL STUDY

There have been many experimental studies of dam break flow conducted in laboratory settings, using very similar experimental arrangements. Classically, laboratory experiments of dam break events involve the rapid vertical removal of a barrier from a flume. Initial experimentation using this method was first published by Schoklitsch [1917b], only providing two results. The experimental study by Eguiazaroff [1935] presented a more complete set of results for a single bed resistance. The experiments conducted by Dressler [1954] were the first complete study of bed resistance on long term dam break flow. The surface profiles produced at dimensionless time $t = 0.5t_{\max}$, where data was captured up to the maximum dimensionless time of $t_{\max} = 300$, were used to calculate a resistance value to produce good agreement between Dressler's analytical model and the experimental results. The results for the entire time period were then compared to his analytical model, however agreement was only found for the long term flow. The experimental results presented by Dressler were used as experimental comparison until the 1990s when new data capturing methods began to be used to record more information about a given flow.

Bell et al. [1992] conducted the first modern experiments on dam break in a straight and curved flume, over a smooth and rough bed for dry and wet tailwater conditions. Bell et al. particularly noted that a dam break wave in a curved channel will have greater elevation and velocity on the outside edge.

Lauber and Hager [1998] presented the first use of Particle image velocimetry (PIV) to capture velocity data from a dam break flow. Lauber and Hager's results show an almost linear increase of fluid velocity from negative wave front to the location of maximum velocity. The maximum fluid velocity coincides with the positive wave front velocity for small time scales. For large time scales, however, the positive front velocity is smaller than the maximum velocity, which is located further back within the flow.

Stansby et al. [1998] studied the surface evolution for the initial stages of dam break flow, noting the differences in wave structure exhibited by dam break over a dry or wet bed. In the case of a wet bed (non-zero tailwater depth), the wave forms a mushroom like shape, caused by the development of positive vertical velocities just ahead of the dam site. This does not occur over a dry bed, where the wave

front progresses along the dry bed with no positive vertical velocity structure in the region beyond the dam site.

In the same year Capart and Young [1998] performed the first experiments into dam break over a mobile, erodible bed. Similar experiments were conducted by Leal et al. [2001] and Fraccarollo and Capart [2002]. The results of these studies vary greatly due to the aforementioned large variation in sediment density and size between these tests. The experiments presented by Fraccarollo and Capart [2002] were soon followed by a series of experiments conducted under the “Sediment Movement” work package of the IMPACT project (Investigation of Extreme Flood Processes and Uncertainty) which resulted in publications by Spinewine and Zech [2003]; Spinewine [2005]; Spinewine and Zech [2007]; Zech et al. [2008]; Soares-Frazão et al. [2012], as well as countless others that have used the produced datasets as a means to validate numerical studies of the dam break problem over mobile beds.

Almost all of the experiments conducted in the study of dam break have used the method of rapid vertical removal of a thin barrier to initiate the laboratory reconstruction of dam break flow, however the rapid tangential movement of a boundary surface is known to impose vorticity on the adjacent fluid. As such this thesis has a focus on the development of a method for the rapid removal of a barrier from a flume in such a way that it does not impose a vortical disturbance on the fluid.

1.5 PRESENT STUDY

The main focus of the present study was the development of a NS solver and fluid tracker to be coupled with a sediment transport model. In order to achieve this, a geometric method was developed to calculate the movement of sediment between the flow and the bed. In addition, a geometric method was developed to redistribute bed heights to enforce the slope limit or critical angle of repose of the sediment. The secondary focus was the generation of experimental data for comparison to the model. The experiments applied new experimental data capturing methods (three dimensional PIV) and new apparatus for a gate removal system (inspired by the methods employed by Dalziel [1993] in the study of Rayleigh-Taylor instability), and sought to validate their use in the study of dam break flows.

1.6 THESIS OUTLINE

In this thesis attention is restricted to two dimensional, laminar, initial stage, dam break flow in a channel with rectangular cross sections of constant width, with and without a bed of mobile, uniform and non-cohesive sediment, with a view to extend the work to al-

low for turbulence. This required a hydrodynamic model, a sediment transport model for the bed and for sediment in suspension, a sink/-source model for erosion and deposition and an avalanching model to account for slope failure within the scour hole.

The thesis is divided into three main chapters. The first chapter is a discussion of the computational model developed to study the dam break problem. There is a review of the methods used to numerically solve the NS Equations, followed by a description of the hybrid solution method developed, which combines various methods, ranks their efficiencies and allows for the solution to be found even if the current method fails to converge. Following this is a description of the free surface model used. This model is based on the Youngs' method 1982, which was not originally fully described. The method presented is an adaptation of the method presented by Rudman [1997], including a variation on the methods used to account for cell rotation. Following this is a description of the sediment transport model used. The model is an amalgamation of some commonly used approaches (Advection diffusion for suspended sediment and the Meyer Peter Müller approach in conjunction with the Exner equation for bed load), however novel methods were developed to geometrically account for the transition between the bed and the suspended load and the redistribution of sediment required to enforce the critical angle of repose slope limit. Finally there is a description of the factors used to limit the time step size to maintain the stability of the simulation.

The second chapter reviews the validity tests used to analyse the correct functioning order of the model's separate parts. Tests were performed separately on the NS solver, the free surface calculator and the sediment transport calculator to determine the limits of their functionality.

The third chapter introduces the experimental procedures used to recreate the dam break problem in the laboratory and compare the results of these tests to data produced by the model described in chapter 2. The final chapter is a discussion of the results, their impact on the hypothesis, and a review of the present model and its limitations.

Appendix A includes supplementary materials including descriptions of other volume fraction tracking methods used for comparison in section 3.2, methods for the conversion of shapes into test objects for section 3.2, descriptions of Ritter's analytical solution and Dressler's analytical solution for dam break used for comparison in section 4.2, documentation for the use of the user interface designed to accompany the simulation and a full listing of the pertinent algorithms.

NUMERICAL MODEL

The creation of the numerical model involved three distinct problems. Firstly the motion of the fluid must be calculated. Secondly, if a free surface exists, the change in the free surface profile must be calculated. Finally, if sediment is to be included, the bed morphology and suspended sediment motion must be calculated.

2.1 FLUID MODEL

The construction of any hydrodynamic model requires a decision to be made on which governing equations are to be solved, in how many dimensions and using what numerical scheme. Vertical accelerations and velocities dominate during the initial stages of dam break flow, which make the use of the depth averaged shallow water equations inappropriate. As such, the 2D Navier-Stokes equations were chosen as they allow for those vertical accelerations and velocities. Simulation in three dimensions was deemed to be unnecessarily computationally intensive, especially since the flow in question is two dimensional in nature. The most common numerical schemes are the Finite Differences Method, the Finite Volume Method and the Finite Element Method. Implementations of Finite Differences tend to be the least complex, whereas both the Finite Volume Method and Finite Element Method tend to be significantly more complex to understand, implement and code, but have the advantages that they are able to be applied to unstructured non uniform meshes that are deformable and are able to apply boundary conditions non-invasively.

The Finite differences method was chosen as the numerical scheme particularly for its reduced complexity and as such reduced barrier to entry, knowing that the main focus of innovation would be in the development of new sediment models.

2.1.1 *Navier-Stokes Solution*

The motion of fluids are governed by the [NS](#) equations, the combination of Newton's second law and the principle of conservation of mass, written for the context of fluid motion. The Eulerian, incom-

pressible [NS](#) equations in two dimensions, separated into the component directions are,

$$\begin{aligned} \left(\frac{\partial u'}{\partial t'} + u' \frac{\partial u'}{\partial x'} + v' \frac{\partial u'}{\partial y'} \right) \rho' &= \mu' \left(\frac{\partial^2 u'}{\partial x'^2} + \frac{\partial^2 u'}{\partial y'^2} \right) - \frac{\partial p'}{\partial x'} + g'_x \rho', \\ \left(\frac{\partial v'}{\partial t'} + u' \frac{\partial v'}{\partial x'} + v' \frac{\partial v'}{\partial y'} \right) \rho' &= \mu' \left(\frac{\partial^2 v'}{\partial x'^2} + \frac{\partial^2 v'}{\partial y'^2} \right) - \frac{\partial p'}{\partial y'} + g'_y \rho', \\ \frac{\partial u'}{\partial x'} + \frac{\partial v'}{\partial y'} &= 0, \end{aligned}$$

where u' is the horizontal component velocity, v' is the vertical component velocity, t' is the time, x' is horizontal coordinate direction, y' is vertical coordinate direction, p' is the pressure, ρ' is the fluid density, μ' is the dynamic viscosity and g'_x and g'_y are the horizontal and vertical externally imposed body accelerations. Note that each of the variables are dimensional, which is denoted by an apostrophe. The standard values for the body accelerations are $g'_x = 0 \text{ms}^{-2}$ and $g'_y = -g' = -9.81 \text{ms}^{-2}$ and the fluid is assumed to be incompressible with a constant fluid density of 1000kgm^{-3} for water.

In an effort to non-dimensionalise the [NS](#) equations, the following non-dimensional ([ND](#)) variables are defined.

$$\begin{aligned} u &= \frac{u'}{U'} & v &= \frac{v'}{U'} & t &= \frac{t' U'}{L'} \\ x &= \frac{x'}{L'} & y &= \frac{y'}{L'} & p &= \frac{p'}{P'} \\ P' &= \rho' U'^2 & g_x &= \frac{g'_x L'}{U'^2} & g_y &= \frac{g'_y L'}{U'^2} \end{aligned}$$

where u is the [ND](#) horizontal component velocity, v is the [ND](#) vertical component velocity, t is the [ND](#) time, x is the [ND](#) horizontal length, y is the [ND](#) vertical length, p is the [ND](#) pressure, g_x and g_y are [ND](#) horizontal and vertical externally imposed [ND](#) body accelerations, L' is a length scale, U' is a velocity scale and P' is a pressure scale. Note that an inertial pressure scale was chosen over a viscous pressure scale ($P' = \mu' U' / L'$) as the flows to be analysed are expected to be fast moving, where inertia will be dominant (Reynolds number will be large, $Re \gg 1$, where $Re = \rho' U' L' / \mu'$).

Using these variables the following [ND](#) form of the Navier-Stokes equations can be derived.

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{Re} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + g_x, \quad (2)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \frac{1}{Re} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + g_y, \quad (3)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \quad (4)$$

The advective terms $u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y}$ in the x-direction and $u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y}$ in the y-direction are replaced with $\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y}$ and $\frac{\partial v^2}{\partial y} + \frac{\partial uv}{\partial x}$ respectively. This allows for a simpler discretisation scheme on a staggered grid. These terms can be seen to equate by expanding the new form using the product rule and applying the continuity equation (4) to cancel out the additional terms.

By discretising the time derivative of equations 2 and 3 using a forward finite difference approximation, the following equations are derived.

$$u^{n+1} = u^n + \delta t \left[\frac{1}{\text{Re}} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \frac{\partial u^2}{\partial x} - \frac{\partial uv}{\partial y} + g_x - \frac{\partial p}{\partial x} \right], \quad (5)$$

$$v^{n+1} = v^n + \delta t \left[\frac{1}{\text{Re}} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - \frac{\partial v^2}{\partial y} - \frac{\partial uv}{\partial x} + g_y - \frac{\partial p}{\partial y} \right]. \quad (6)$$

where n is an index denoting the current time-step and δt is the ND time-step length.

Chorin's projection method [1968], a predictor-corrector method, is used to calculate values of u , v and p for the time after one time step. This is achieved by removing the pressure term from equations 5 and 6 to calculate a prediction for the component velocities. The pressure is then separately found by the solution of a set of linear equations, see section 2.1.2. The newly calculated pressure values are then used to correct the prediction for the component velocities. Equations 7 and 8 represent the prediction step and equations 9 and 10 represent the application of the correction, where the values for p are calculated separately. Thus the prediction and correction equations are

$$F = u^n + \delta t \left[\frac{1}{\text{Re}} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \frac{\partial u^2}{\partial x} - \frac{\partial uv}{\partial y} + g_x \right], \quad (7)$$

$$G = v^n + \delta t \left[\frac{1}{\text{Re}} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - \frac{\partial v^2}{\partial y} - \frac{\partial uv}{\partial x} + g_y \right], \quad (8)$$

$$u^{n+1} = F - \delta t \frac{\partial p}{\partial x}, \quad (9)$$

$$v^{n+1} = G - \delta t \frac{\partial p}{\partial y}, \quad (10)$$

where F and G are the predictions to the component velocities u and v respectively.

The spatial discretisation scheme is accomplished by employing a staggered grid, where a cell's horizontal velocity u is defined at the centre of the right hand cell edge, the vertical velocity component v is defined at the centre of the upper cell edge and the pressure is defined at the cell centre. A staggered grid was employed over a

collocated grid, where the velocity and pressure terms are found at a single point for each cell, as collocated grids are known to cause oscillatory instabilities [Griebel et al., 1998]. See figure 4 for an illustration of the location and indexing system used to refer velocity and pressure for the cell (i, j) .

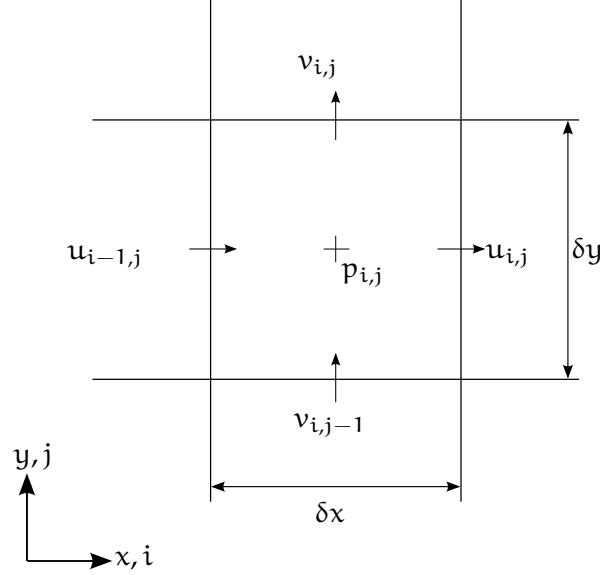


Figure 4: Cell illustration showing the staggered location of the velocity and pressure and the cell indexing system used throughout the thesis. Note that the horizontal velocity prediction F is collocated with u and the vertical velocity prediction G is collocated with v

The choice of discretisation scheme for the advective terms can have a large effect on both the stability and the error of a numerical scheme, especially where advection is dominant. The standard central difference approximation can cause an oscillatory instability, however this instability can be avoided by using an upwind or donor-cell scheme [Griebel et al., 1998; Ferreira et al., 2002]. When discretising the differential term $\frac{\partial(ku)}{\partial x}$, an upwind or donor-cell scheme uses the sign of the donor-cell term k to choose whether a backward or forward difference is applied. In the advective case presented, the donor-cell term k is equal to the mean velocity across the cells either side of the cell in question, and the sign of this term denotes the direction of the flow in the neighbouring cells. The upwind or donor-cell method, however, should suffer a reduction in accuracy as it is only a first order approximation. The present solver employs a hybrid first order-second order scheme for the approximation of the advective terms, a weighted combination of a half width central difference and the donor-cell scheme. This approach was notably employed by Hirt et al. [1975] and Griebel et al. [1998] and is based on the work pre-

sented by Gentry et al. [1966]. Refer to Ferreira et al. [2002] for further discussion on the upwind method and alternative approximations for the advective terms. A second order central difference approximation is employed for the viscosity terms.

The spatial discretisations used for the calculation of the horizontal component velocity prediction, F , are as follows.

$$\begin{aligned} \frac{\partial (u^2)}{\partial x} &= \frac{1}{\delta x} \left[\left(\frac{u_{i,j} + u_{i+1,j}}{2} \right)^2 - \left(\frac{u_{i-1,j} + u_{i,j}}{2} \right)^2 \right] \\ &\quad + \gamma \frac{1}{\delta x} \left[\frac{|u_{i,j} + u_{i+1,j}|}{2} \frac{(u_{i,j} - u_{i+1,j})}{2} \right. \\ &\quad \left. - \frac{|u_{i-1,j} + u_{i,j}|}{2} \frac{(u_{i-1,j} - u_{i,j})}{2} \right], \\ \frac{\partial uv}{\partial y} &= \frac{1}{\delta y} \left[\frac{(v_{i,j} + v_{i+1,j})}{2} \frac{(u_{i,j} + u_{i,j+1})}{2} \right. \\ &\quad \left. - \frac{(v_{i,j-1} + v_{i+1,j-1})}{2} \frac{(u_{i,j-1} + u_{i,j})}{2} \right] \\ &\quad + \gamma \frac{1}{\delta y} \left[\frac{|v_{i,j} + v_{i+1,j}|}{2} \frac{(u_{i,j} - u_{i,j+1})}{2} \right. \\ &\quad \left. - \frac{|v_{i,j-1} + v_{i+1,j-1}|}{2} \frac{(u_{i,j-1} - u_{i,j})}{2} \right], \\ \frac{\partial^2 u}{\partial x^2} &= \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\delta x^2}, \\ \frac{\partial^2 u}{\partial y^2} &= \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\delta y^2}, \\ \frac{\partial p}{\partial x} &= \frac{p_{i+1,j} - p_{i,j}}{\delta x}, \end{aligned}$$

where i is an index denoting a cell's horizontal location, j is an index denoting a cell's vertical location, δx is the ND cell width, δy is the ND cell height and γ is the donor-cell weighting factor. Note that a donor-cell weighting factor of zero recovers the central difference and a weighting factor of one recovers an upwind or donor-cell scheme.

The discretisations used for the calculation of the vertical component velocity predictions, G , are as follows.

$$\begin{aligned} \frac{\partial (v^2)}{\partial y} &= \frac{1}{\delta y} \left[\left(\frac{v_{i,j} + v_{i,j+1}}{2} \right)^2 - \left(\frac{v_{i,j-1} + v_{i,j}}{2} \right)^2 \right] \\ &\quad + \gamma \frac{1}{\delta y} \left[\frac{|v_{i,j} + v_{i,j+1}|}{2} \frac{(v_{i,j} - v_{i,j+1})}{2} \right. \\ &\quad \left. - \frac{|v_{i,j-1} + v_{i,j}|}{2} \frac{(v_{i,j-1} - v_{i,j})}{2} \right], \end{aligned}$$

$$\begin{aligned} \frac{\partial uv}{\partial x} = \frac{1}{\delta x} & \left[\frac{(u_{i,j} + u_{i,j+1})}{2} \frac{(v_{i,j} + v_{i+1,j})}{2} \right. \\ & \left. - \frac{(u_{i-1,j} + u_{i-1,j+1})}{2} \frac{(v_{i-1,j} + v_{i,j})}{2} \right] \\ & + \gamma \frac{1}{\delta x} \left[\frac{|u_{i,j} + u_{i,j+1}|}{2} \frac{(v_{i,j} - v_{i+1,j})}{2} \right. \\ & \left. - \frac{|u_{i-1,j} + u_{i-1,j+1}|}{2} \frac{(v_{i-1,j} - v_{i,j})}{2} \right], \end{aligned}$$

$$\begin{aligned} \frac{\partial^2 v}{\partial x^2} &= \frac{v_{i+1,j} - 2v_{i,j} + v_{i-1,j}}{\delta x^2}, \\ \frac{\partial^2 v}{\partial y^2} &= \frac{v_{i,j+1} - 2v_{i,j} + v_{i,j-1}}{\delta y^2}, \\ \frac{\partial p}{\partial y} &= \frac{p_{i,j+1} - p_{i,j}}{\delta y}. \end{aligned}$$

A stability condition must be defined for the discretisation scheme used for the advection terms. The donor-cell weighting factor, γ , is chosen such that γ is larger than the proportion of the cell a fluid could move through during the time step, whilst not being so large so as to introduce unnecessary error [Hirt et al., 1975]. The conditions are

$$\begin{aligned} 1 &> \gamma > |u_{\max}| \frac{\delta t}{\delta x}, \\ 1 &> \gamma > |v_{\max}| \frac{\delta t}{\delta y}. \end{aligned}$$

where u_{\max} and v_{\max} are the maximum values in the horizontal and vertical velocity fields. Thus γ is

$$\gamma = \tau_\gamma \max \left(|u_{\max}| \frac{\delta t}{\delta x}, |v_{\max}| \frac{\delta t}{\delta y} \right),$$

where τ_γ is the gamma safety factor, a value set to ensure that γ remains larger than the values from which it is defined. The value of τ_γ is usually set within the range [1.2 1.5], a rule of thumb suggested by Hirt and Nichols [1981]. The effect of changing both γ and τ_γ were investigated, the results of which are discussed in full in section 3.1.

The calculation of γ must occur at the beginning of each time step, directly after the calculation of the new time step size, δt . See section 2.4 for details on the dynamic calculation of the time step size.

2.1.2 Solving the Poisson equation for pressure

The correction of the initial solution for the velocity is dependant on the calculation of the pressure field for the following time-step. This

is accomplished by differentiating equations 9 and 10 in their respective directions x and y , and substituting the result into the continuity equation (4). This results in the Pressure Poisson Equation (PPE), 11, which can generate a set of linear equations that can be solved iteratively for the pressure term.

$$\frac{\partial^2 p^{n+1}}{\partial x^2} + \frac{\partial^2 p^{n+1}}{\partial y^2} = \frac{1}{\delta t} \left(\frac{\partial F^n}{\partial x} + \frac{\partial G^n}{\partial y} \right) \quad (11)$$

The set of linear equations is created by discretising the pressure terms with a 2nd order central finite difference approximation and the velocity prediction terms with a 1st order backward finite difference approximation, see equation 12.

$$\begin{aligned} \frac{p_{i+1,j}^{n+1} - 2p_{i,j}^{n+1} + p_{i-1,j}^{n+1}}{\delta x^2} + \frac{p_{i,j+1}^{n+1} - 2p_{i,j}^{n+1} + p_{i,j-1}^{n+1}}{\delta y^2} \\ = \frac{1}{\delta t} \left(\frac{F_{i,j}^n - F_{i-1,j}^n}{\delta x} + \frac{G_{i,j}^n - G_{i,j-1}^n}{\delta y} \right) \end{aligned} \quad (12)$$

The various iterative methods employed by the model to solve equation 12 and the method used to determine convergence are described below.

2.1.2.1 Determining Convergence

Convergence is determined by the calculation of the residual error, r^{it} , where it is the current iteration. The residual error field is calculated by subtracting the right hand side from the left hand side of equation 12 for each cell. As the solution converges, this value should tend toward zero. The equation for the residual error is

$$r_{i,j}^{it} = \frac{p_{i+1,j}^{it} - 2p_{i,j}^{it} + p_{i-1,j}^{it}}{\delta x^2} + \frac{p_{i,j+1}^{it} - 2p_{i,j}^{it} + p_{i,j-1}^{it}}{\delta y^2} - \text{rhs}_{i,j}, \quad (13)$$

where rhs is the right hand side of equation 12. For simple comparison, it is useful to have a single measure of the size of the error for a given pressure field. The L^2 -norm ($\|r^{it}\|_2$) is employed as the comparative measure for the present solver, and is defined as

$$\|r^{it}\|_2 = \left(\frac{1}{i_{\max} j_{\max}} \sum_{i=1}^{i_{\max}} \sum_{j=1}^{j_{\max}} (r_{i,j}^{it})^2 \right)^{\frac{1}{2}}, \quad (14)$$

where $i_{\max} = n_{\text{cols}} - 2$, $j_{\max} = n_{\text{rows}} - 2$, n_{cols} is the number of columns and n_{rows} is the number of rows, including ghost rows and columns. Ghost rows and columns are placed around the grid to impose boundary conditions, see section 2.1.3. Note that equation 14 is correct for an indexing system where a row goes from 0 to $n_{\text{cols}} - 1$ and a column goes from 0 to $n_{\text{rows}} - 1$.

2.1.2.2 Gauss-Seidel

The simplest iterative solution method employed by the solver is the Gauss-Seidel (GS) method. The order in which the method's equation is applied to the grid will have an effect as the equation references the pressures of the surrounding cells, which may already have been overwritten with new values. Two versions of the method are used, a standard GS method and the Red-Black Gauss-Seidel (RBCS) method.

The standard method applies equation 15 to each cell sequentially, moving from left to right within a row for all the rows from the bottom of the grid to the top. In the standard method, the pressure below and to the left of any normal internal cell will have already been overwritten by the algorithm.

The RBCS method uses the same equation, however it applies it in a chessboard fashion, applying the equation first to all the cells representing one colour on the chessboard and then to the cells representing the other colour, see figure 5. Thus, all the surrounding pressures referenced by any cell on the first colour sweep will not have been overwritten and all the surrounding pressures on the second colour sweep will have been overwritten by the first sweep. The GS equation for the solution of the Poisson equation for pressure (12) is

$$p_{i,j} = \frac{\frac{p_{i,j+1} + p_{i,j-1}}{\delta x^2} + \frac{p_{i+1,j} + p_{i-1,j}}{\delta y^2} - \text{rhs}_{i,j}}{\frac{2.0}{\delta x^2} + \frac{2.0}{\delta y^2}} \quad (15)$$

where rhs is the right hand side of equation 12.

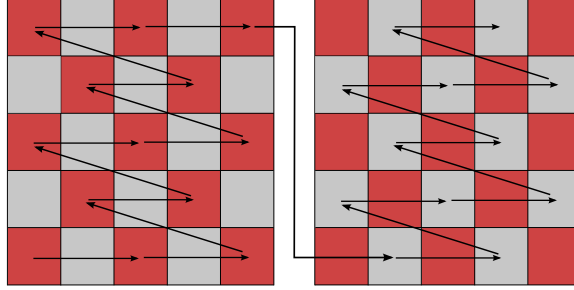


Figure 5: Red-Black Gauss-Seidel

2.1.2.3 Successive Over Relaxation

Successive over-relaxation (SOR) is a variation of the GS method which can see an improvement in convergence speed. The SOR equation for the solution of the PPE successive over relaxation is

$$p_{i,j} = (1 - \omega)p_{j,i} + \frac{\omega}{\frac{2.0}{\delta x^2} + \frac{2.0}{\delta y^2}} \left(\frac{p_{i,j+1} + p_{i,j-1}}{\delta x^2} + \frac{p_{i+1,j} + p_{i-1,j}}{\delta y^2} - \text{rhs}_{i,j} \right), \quad (16)$$

where ω is a relaxation factor, which, for values between 0 and 2, will guarantee eventual convergence for symmetric, positive definite matrices. When $0 < \omega < 1$, the method is referred to as under-relaxation and when $1 < \omega < 2$, the method is referred to as over-relaxation. Note that the equation reduces to the GS method for a relaxation factor of $\omega = 1$. The value of the relaxation factor that allows for the fastest convergence is situation dependant and its computation is non trivial. When no initial estimate for the relaxation factor is known, the Hirt et al. [1975] value of 1.8 is prescribed.

2.1.2.4 Conjugate Gradients

The iterative conjugate gradients (CG) method developed by Hestenes and Stiefel [1952] can significantly reduce the number of steps to convergence compared to the GS, RBGS and SOR methods. The CG method, however, can be less numerically stable or robust. This can lead to the method failing to converge for some situations. The specific equations used for the solution of the PPE using the iterative conjugate gradients method are given below. Refer to Hestenes and Stiefel [1952] and Saad [2003] for further information on the method and its derivation. The method uses a conjugate vector to calculate the pressure solution, where vectors (φ_i, φ_k) are conjugate if they obey $\varphi_i^T \mathbf{A} \varphi_k = 0$. The matrix \mathbf{A} is in this case the matrix created by the left hand side of equation 12. The initial condition for the method is to set the conjugate vector φ to equal the residual error for the first iteration, $\varphi_{i,j}^0 = r_{i,j}^0$. The equations for the iterative solution of the PPE by method of conjugate gradients are

$$\begin{aligned} \text{r2sum}^{\text{it}} &= \sum_{i,j=0}^{i_{\max}, j_{\max}} (r_{i,j}^{\text{it}})^2 \\ \varphi \text{sum}^{\text{it}} &= \sum_{i,j=0}^{i_{\max}, j_{\max}} \left(\frac{\varphi_{i,j}^{\text{it}} \left(\varphi_{i+1,j}^{\text{it}} - 2\varphi_{i,j}^{\text{it}} + \varphi_{i-1,j}^{\text{it}} \right)}{\delta x^2} \right. \\ &\quad \left. + \frac{\varphi_{i,j}^{\text{it}} \left(\varphi_{i,j+1}^{\text{it}} - 2\varphi_{i,j}^{\text{it}} + \varphi_{i,j-1}^{\text{it}} \right)}{\delta y^2} \right) \end{aligned}$$

$$\begin{aligned}
\eta^{it} &= \frac{r2sum^{it}}{\varphi sum^{it}} \\
p_{i,j}^{it+1} &= p_{i,j}^{it} + \eta^{it} \varphi_{i,j}^{it} \\
r_{i,j}^{it+1} &= r_{i,j}^{it} - \eta^{it} \left(\frac{\varphi_{i+1,j}^{it} - 2\varphi_{i,j}^{it} + \varphi_{i-1,j}^{it}}{\delta x^2} \right. \\
&\quad \left. + \frac{\varphi_{i,j+1}^{it} - 2\varphi_{i,j}^{it} + \varphi_{i,j-1}^{it}}{\delta y^2} \right)
\end{aligned}$$

where φ is the conjugate vector, $r2sum$ is the sum of the residual error squared, φsum is equal to $\varphi^T \mathbf{A} \varphi$ and η is the factor applied to find the change in the pressure.

At this point a convergence check is made by calculating the L^2 -norm of the residual error. If convergence has been reached the process ends, otherwise the conjugate vector φ is recalculated (using the equations presented below) for the next iteration and the process returns to the beginning.

$$\begin{aligned}
\xi^{it+1} &= \frac{r2sum^{it+1}}{r2sum^{it}} \\
\varphi_{i,j}^{it+1} &= r_{i,j}^{it+1} + \xi \varphi_{i,j} \\
it &= it + 1
\end{aligned}$$

where ξ is the factor applied to find the change in the values of the conjugate vector for the next time-step.

2.1.2.5 Multigrid method

The multigrid (**MG**) method uses one or more coarsened grids to calculate a converged solution more rapidly than a conventional iterative solution, with fewer robustness issues akin to those of conjugate gradients. The advantages gained by using the **MG** method can be illustrated by considering the errors produced by a method and the error reduction speed for different frequencies of errors. The **GS** method efficiently reduces high frequency or local errors, but is inefficient at reducing low frequency or global errors. This is problematic for the solution of the **PPE** due to the smooth nature of the solution, causing a tendency towards low frequency errors that are slow to be smoothed by conventional iterative solvers. Additionally the increase in grid resolution compounds this effect, so the use of a finer grid causes the solution to be smoother and thus the error to be of lower frequency. The **MG** method uses coarser grids, which transform the error from a low to a high frequency. The **GS** method then becomes very efficient at solving the problem, reducing the error for the reduced number of points. These values can be imposed onto the finer grid, which also transforms the low frequency error into a high frequency error. The simplest **MG** method is a V-cycle. The initial pressure field is restricted to a coarser grid. The values are smoothed using **GS** and a

residual error field is calculated. This field is substituted back into the algorithm, recursively being restricted and smoothed until the coarsest grid is reached. At this point the method calculates an exact solution, equivalent to the use of a single GS step, and then proceeds to prolongate the values onto a finer grid (smoothing using the GS method as it goes) until it reaches the original grid resolution. This method is illustrated in figure 6.

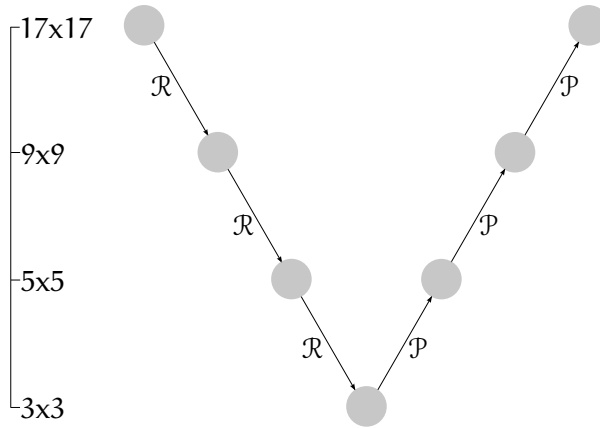


Figure 6: Multigrid V-cycle, where \mathcal{R} denotes restriction from a fine grid to a coarse grid and \mathcal{P} refers to prolongation from a coarse grid to a fine grid. Grid size chosen to be equal to $2n - 1$, where n is equal to the grid size of the previous coarser grid, or, for the coarsest grid, n is equal to 2. Diagram based on figure 20.6.1 page 1070, presented by Press et al. [2007]

Restriction, or the process of simplifying a fine grid's values onto a coarse grid is done by applying a Restriction factor which weights the various surrounding values for their contribution towards the value given at the new coarse grid point. The restriction factor used is the half weighting restriction factor, as given below:

$$\mathcal{R} = \begin{bmatrix} 0 & \frac{1}{8} & 0 \\ \frac{1}{8} & \frac{1}{2} & \frac{1}{8} \\ 0 & \frac{1}{8} & 0 \end{bmatrix}$$

Thus in restriction the value from the fine grid directly above the new coarse grid value is multiplied by 0.5 and added to the fine values directly above, below, to the left and right of the point in question, multiplied by 0.125.

Prolongation is the process by which the grid is interpolated from a coarse grid onto a fine grid. Bilinear interpolation is used to achieve this, using the following prolongation factor:

$$\mathcal{P} = \begin{bmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{bmatrix}$$

Thus prolongation sets the outer points to equal the fine points they lie on top of, where internal points are taken from a weighting of a 3×3 grid. The prolongation and restriction factors are those suggested by the implementation of the method presented by [Press et al. \[2007\]](#). For further information on the derivation of the method refer to [Briggs et al. \[2000\]](#) and for further information on the method's implementation, refer to [Press et al. \[2007\]](#).

Figure 6 illustrates the coarsening process applied to an even or square grid, where the grid dimensions are equal. The MG method can be applied to an uneven grid, however, the coarsest grid solution will not be the solution of a single cell. For example, if one grid dimension is one level higher than the other dimension (i.e. the grid is twice as wide as it is high or twice as high as it is wide) then the coarsest grid will be 5×3 as opposed to 3×3 , meaning there will be three internal cells as opposed to there being only one. This is not a problem as the solution of the coarsest grid is enacted by a smoothing step of GS, and this can be applied to the uneven coarsest grid, and then proceed to move back up to the finer grid levels.

2.1.2.6 A progressive Solution

The various solution methods given have different levels of robustness, convergence speed (number of iterations) and computational speed (time taken to compute). In an attempt to more efficiently choose the most appropriate method a progressive solution was defined. The progressive solution, for the first time step, will run all of the solution methods, outputting the final L^2 norm value, number of iterations, and time taken to compute for each method. Using these values, a ranking algorithm orders the methods, and in the case that the solution method fails, the next solution method in the sorted list will be used.

When a method has met the convergence criterion, the number of iterations for that method will be less than the maximum allowable number of iterations. If this is true for all methods, the time taken to compute is compared for all methods, with the smallest time ranked first and the largest time ranked last. If the number of iterations reached the maximum value for any of the methods, the list of methods is first sorted by time to compute, the fastest of which is

deemed the base method. Then a measure of the size of the improvement in L^2 norm with respect to the additional time taken to compute is calculated, where improvement in L^2 norm and additional time to compute are comparisons to the base method. The equation for this calculated rank measure is

$$\text{rank} = \frac{\|r^{it}\|_{2,\text{base}} - \|r^{it}\|_{2,\text{new}}}{t_{c,\text{new}} - t_{c,\text{base}}} \quad (17)$$

where t_c is the time taken to compute, the subscript base refers to the method with the shortest computational time and the subscript new refers to the current method that is being compared. The base method is given a calculated rank of zero. Methods that are more efficient than the base method will be positive and methods less efficient than the base method will be negative. Thus the list of methods can be sorted by their calculated rank, using the method with the largest calculated rank until failure necessitates the use of a different method, at which point the method with the next greatest calculated rank will be used.

Additionally, the algorithm allows for the recalculation of the efficiency ranking list after some given time or time-step interval, to ensure the list order remains accurate and the most efficient method is being used primarily. The interval must be maximised, such that the additional computational cost associated with the recalculation of the list does not negate the improvements gained from using the most efficient solution method.

2.1.3 Boundary Conditions

Boundary conditions are applied to the grid by applying a single layer of ghost cells around the outside of the grid. This is illustrated in figure 7. The model also allows for arbitrarily shaped regions through the application of a mask that sets cells outside of the region as boundary cells which apply the following boundary conditions.

Four boundary conditions are defined for use in the present model: No slip, Free slip, Variable bed and Periodic. All rigid boundaries are first given a no penetration condition, where perpendicular velocities are set to zero at the boundary. Due to the staggered nature of the velocity field, the perpendicular velocities can be directly prescribed. For example at a left hand vertical boundary, the perpendicular velocity collocates with the horizontal velocity of the ghost cell as such we can set $u_{0,j} = 0$. A no slip condition sets the tangential velocities to equal to zero at the boundary edge. This condition must be indirectly prescribed, by setting the tangential velocity of the ghost cell, which is offset from the boundary edge, see figure 8. A free slip condition sets the tangential velocity to equal the nearest parallel component velocity, see figure 9. A dynamic bed boundary condition

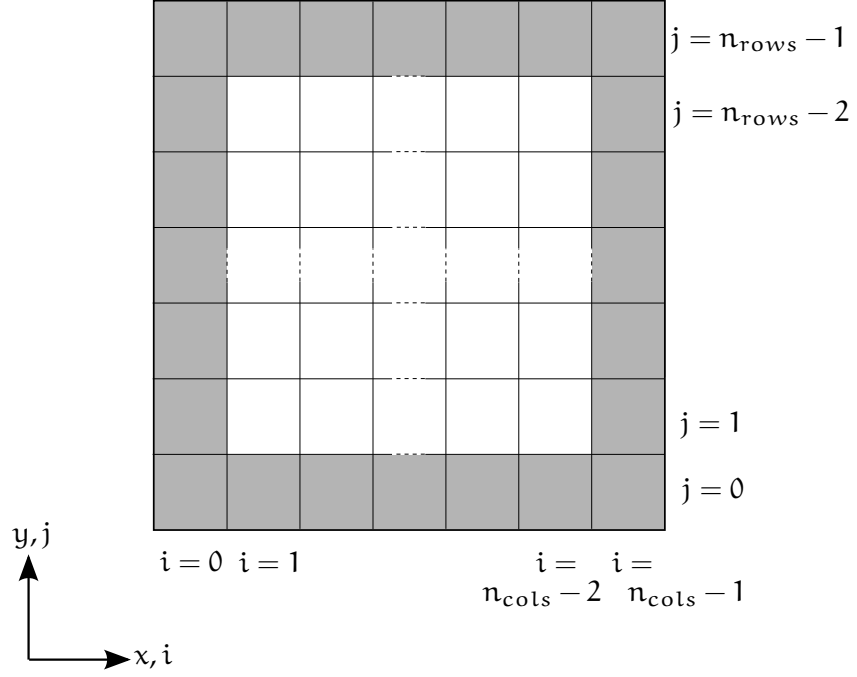


Figure 7: Grid illustration

accounts for the variable nature of a dynamic sediment bed height. No slip and free slip variable bed boundary conditions are provided. For both cases a no penetration condition is maintained by prescribing a linear relationship to set the perpendicular velocity of the cell. Tangential velocities in the no slip case are also calculated via a linear relationship and tangential velocities for the free slip are imposed by equating the velocity in the cell to the nearest parallel, as in the rigid bed case. See figure 12 for an illustration of the dynamic bed and the equations for the calculation of the velocity boundary conditions at a variable bed.

Finally, a periodic boundary condition is a non-rigid boundary condition, where one boundary acts as a direct connection to the boundary on the opposite side. This is accomplished by setting the velocities of the ghost cells at one boundary to equal the velocities of the real cell bordering the opposite boundary (see figures 10 and 11)

Two boundary conditions are prescribed for the pressure. The pressure gradient across a rigid boundary is assumed to be equal to zero. Using this condition, the pressure in any ghost or bed cell is set to equal the pressure in its neighbouring cell real cell, for example in the case of a rigid left hand boundary, the pressure is set as $p_{0,j} = p_{1,j}$. The second boundary condition for the pressure is a periodic boundary condition, where the pressure is set to equal the pressure of the

real cell that neighbours the equivalent boundary cell on the opposite boundary. Note that the iterative methods used to solve the PPE also require a boundary condition for the velocity predictions F and G . As these are velocities and align with the perpendicular or normal velocity boundary condition, the velocity predictions are set to equal the velocities at those points.

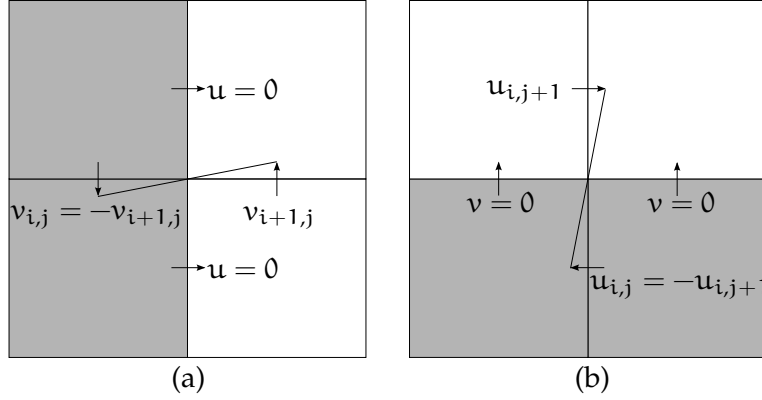


Figure 8: No slip boundary conditions, (a) vertical boundary at the left hand side and (b) horizontal boundary at the lower side

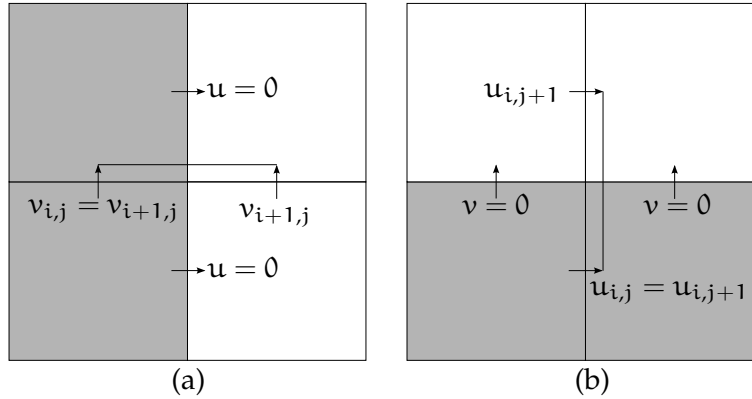


Figure 9: Free slip boundary conditions, (a) vertical boundary at the left hand side and (b) horizontal boundary at the lower side

2.2 FREE SURFACE TRACKING

In the simulation of fluids with free surfaces, as is the case for dam break flows, the position and movement of the free surface must be tracked from step to step.

Various methods have been developed over the years to enable free surface tracking, and they are broadly separated into three categories: Surface tracking, Volume tracking and Movable Mesh Methods [Hyman, 1984]. Surface tracking defines specific marker points on the surface, and interpolates a surface between those points. Volume track-

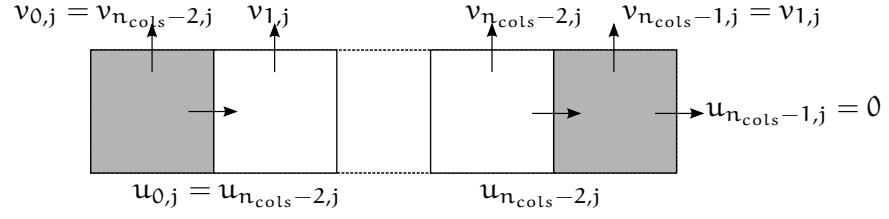


Figure 10: Periodic boundary condition, horizontally across the grid

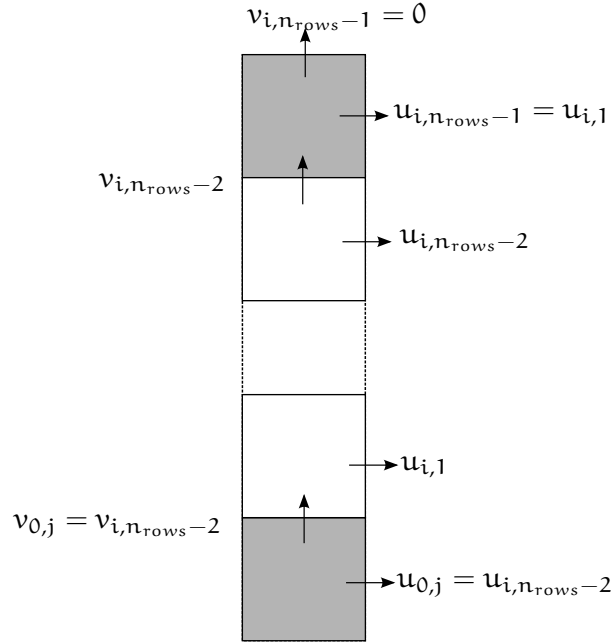


Figure 11: Periodic boundary condition, vertically across the grid

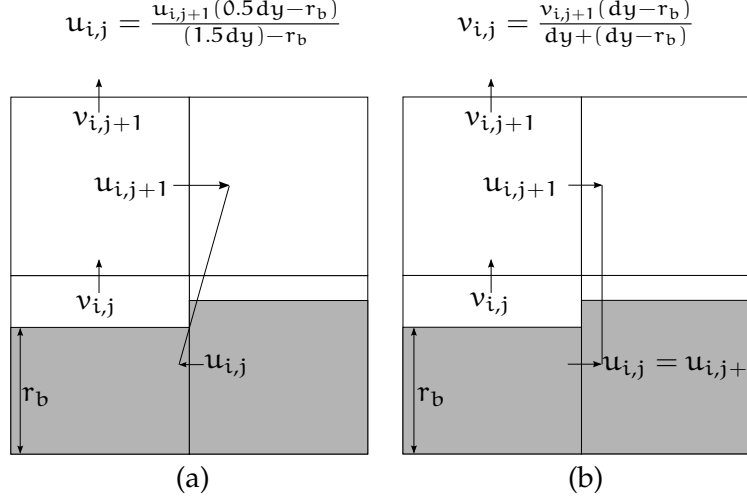


Figure 12: Dynamic bed boundary condition, where r_b is the bed height within the highest bed cell, or the bed cell remainder. (a) Horizontal component velocity boundary condition, (b) vertical component velocity boundary condition for r_b values not equal to zero ($v_{i,j} = 0$ when $r_b = 0$)

ing methods are usually subdivided into Volume Fraction tracking or Volume of Fluid (VOF) methods and Marker and Cell methods. Volume fraction tracking methods track the movement of the free surface by tracking the volume of fluid that exists in each cell, whereas Marker and Cell methods track the location of the fluid by tracking the location of marker particles that are dispersed throughout the fluid. Finally, Movable Mesh Methods allow for the placement of the interface in question at cell boundaries through the deformation of the mesh (a method particularly useful for Finite Volume and Finite Element models).

The primary method used by the current solver to track the fluid in the system is an adapted implementation of the Youngs' Volume of Fluid (YVOF) method, a volume fraction tracking method developed by Youngs [1982] to supplant simpler linear or stair stepped interface methods.

In addition to Youngs' method, the solver also has the capability to run simulations using the Simple Marker and Cell (SMAC) method [Amsden and Harlow, 1970], the Simple Linear Interface Calculator (SLIC) method [Noh and Woodward, 1976] and Chorin's Flame Advection and Propagation (FLAP) method [Chorin, 1980]. The volume fraction tracking methods are compared as part of the validity testing presented in section 3.2. For further details on the alternate tracking methods, refer to appendix A.

The YVOF method [Youngs, 1982], is a volume fraction tracking method. The volume fraction, c , is the proportion of a cell's volume (or area in two dimensions) that is filled with a specific fluid and as

such ranges from a value of 0, where the cell is empty, to 1, where the cell is full. Cells that have a volume fraction between 0 and 1 are said to contain a free surface or interface. Youngs' method calculates an approximate interface angle and effectively moves it within the cell until the area of the geometry created equals the area equivalent to the volume fraction of the cell. The method was not fully described in the original paper, however an interpretation of the method was made by Rudman [1997]. The method presented below is a formulation of the method presented by Rudman, correcting some minor mistakes and creating a full procedure to account for the various rotations of the four basic cell types given by Rudman, as seen in figure 11 on page 690 presented in Rudman [1997] or below in row A of figure 13.

Youngs' method can be split into three parts, identification of fluid geometry, calculation of volume fluxes and advection of volume fraction.

2.2.1 Geometry identification

To identify the geometry of the fluid in a given cell we must calculate the angle of the interface, discover which combination of cell edges the interface intersects (case selection), calculate the proportion of those cell edges which are in contact with fluid and find the orientation of the cell.

The method first requires the calculation of an approximate surface angle, β , and a square cell equivalent angle, α . These angles are calculated by applying a 3x3 stencil, originally used by Kothe et al. [1991], to the volume fractions in the eight surrounding cells.

$$\begin{aligned}
 n_{i,j}^x &= \frac{1}{\delta x} (c_{i+1,j+1} + 2c_{i+1,j} + c_{i+1,j-1} \\
 &\quad - c_{i-1,j+1} - 2c_{i-1,j} - c_{i-1,j-1}) \\
 n_{i,j}^y &= \frac{1}{\delta y} (c_{i+1,j+1} + 2c_{i,j+1} + c_{i-1,j+1} \\
 &\quad - c_{i+1,j-1} - 2c_{i,j-1} - c_{i-1,j-1}) \\
 \beta &= \tan^{-1} \left(\frac{-n^x}{n^y} \right) \\
 \alpha &= \tan^{-1} \left(\frac{\delta x}{\delta y} \tan \beta \right)
 \end{aligned}$$

The values of both β and α lie in the range $[-\pi/2, \pi/2]$. The case selection process requires α to be in the range $[0, \pi/2]$. In the case that the angles generated lie between 0 and $-\frac{\pi}{2}$, a 90 degree rotation is applied to this angle by calculating $\alpha = \frac{\pi}{2} - |\alpha|$. β is simply made positive ($\beta = |\beta|$) as the volume flux calculation deals with the cell in its original un-rotated orientation.

The next stage is to determine which of the cell edges the interface intersects. The interface can go from the bottom edge to the right edge (case I), from the left edge to the right edge (case II), from the bottom edge to the top edge (Case III) or from the left edge to the top edge (case IV). Refer to algorithm 1 for the full case selection procedure. Note that when algorithm 1 refers to scenario B/D, it is referring to the cell scenarios that are generated by an alpha value that was originally negative. The exception referred to is only in effect during the flux calculation stage and is further discussed in section 2.2.2.

Algorithm 1 : Case Selection for s-value calculation and flux calculation

Input : Volume fraction field, c , approximate surface angle,
 $0 \leq \alpha \leq \frac{\pi}{2}$
Output : Case number

```

1  forall the Fluid surface cells do
2      if  $\alpha < \pi/4$  then
3          if  $c \leq \frac{1}{2} \tan \alpha$  then
4              Case I
5          else if  $c \leq 1 - \frac{1}{2} \tan \alpha$  then
6              Case II (or case III for scenario B/D during flux
                        calculation)
7          else
8              Case IV
9          end
10     else
11         if  $c \leq \frac{1}{2} \cot \alpha$  then
12             Case I
13         else if  $c \leq 1 - \frac{1}{2} \cot \alpha$  then
14             Case III (or case II for scenario B/D during flux
                        calculation)
15         else
16             Case IV
17         end
18     end
19 end

```

Once the case has been determined, the geometry of the fluid in the cell is calculated via the definition of the filled edge proportions or s-values, s_l , s_r , s_t and s_b , where l,r,t, and b refer to the left hand, right hand, top and bottom cell edges respectively. The s-values represent the proportion of a cell edge in contact with the fluid in the cell. The combination of these s-values indicate the location of the intersection between the interface and the cell edges.

| Case I | Case II |
|--------------------------------------|--|
| $s_T = 0$ | $s_T = 0$ |
| $s_R = \sqrt{2c_{i,j} \tan(\alpha)}$ | $s_R = c_{i,j} + 0.5 \tan(\alpha)$ |
| $s_B = \sqrt{2c_{i,j} \cot(\alpha)}$ | $s_B = 1$ |
| $s_L = 0$ | $s_L = c_{i,j} - 0.5 \tan(\alpha)$ |
| Case III | Case IV |
| $s_T = c_{i,j} - 0.5 \cot(\alpha)$ | $s_T = 1 - \sqrt{2(1 - c_{i,j}) \cot(\alpha)}$ |
| $s_R = 1$ | $s_R = 1$ |
| $s_B = c_{i,j} + 0.5 \cot(\alpha)$ | $s_B = 1$ |
| $s_L = 0$ | $s_T = 1 - \sqrt{2(1 - c_{i,j}) \tan(\alpha)}$ |

The equations presented for Case IV, include a correction to those originally presented by Rudman, where the $(1 - c_{i,j})$ term had erroneously been given simply as $c_{i,j}$.

Finally the orientation of the cell must be defined. There are four possible orientations, each of which are applicable to the four cases. The four original cases described above, where the fluid is found on the right hand side of the interface, are referred to as Orientation A. These cases rotated by 90° are referred to as orientation B, by 180° are referred to as orientation C and by 270° are referred to as orientation D. Figure 13 illustrates the four Orientations. Orientations A and C have a positive original value of α , orientations B and D have a negative original value of α . The distinction between A and C (or B and D) can be found by studying the stencils used to calculate the angles to determine whether there is more fluid above or below the cell or whether there is more fluid to the left or the right of the cell.

At this point, knowing the orientation, case and the s-values it is useful to consider the cells to have rotated back to their original positions. Refer to table 2 for the rotations to be applied to the s-values.

| | A | B | C | D |
|-------|-------|-------|-------|-------|
| s_T | s_T | s_L | s_B | s_R |
| s_R | s_R | s_T | s_L | s_B |
| s_B | s_B | s_R | s_T | s_L |
| s_L | s_L | s_B | s_R | s_T |

Table 2: Rotation of Filled edge proportions or s-values to match their orientations

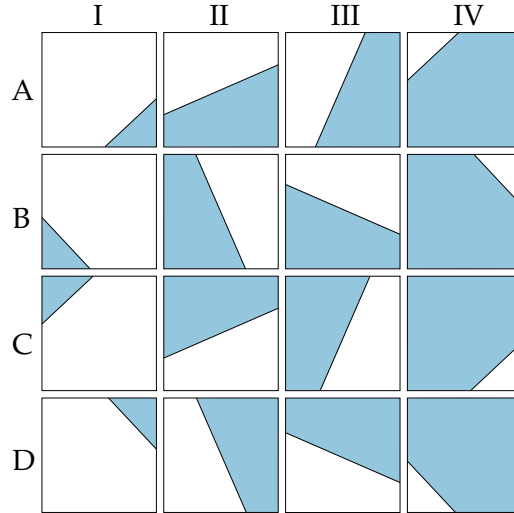


Figure 13: Orientations (A-D) and Cases (I-IV) for Youngs' method. Diagram based on figure 11 page 690 presented in Rudman [1997]

2.2.2 Volume flux calculation

In order to calculate the change in the free surface profile, we must first calculate the geometry of the free surface, we can then use the velocity fields to calculate volume that will move into and out of each cell, also known as the volume flux. We define the volume fluxes F_l , F_r , F_b and F_t as the outward volume fluxes moving across the left hand, right hand, bottom and top cell edge respectively. These fluxes only exist in the circumstance that the fluid velocity at the cell edge points outwards (positive for the right hand edge, negative for the left hand edge etc.), otherwise the flux is set to zero. In the case that the velocity is pointing inwards, the flux calculation for the neighbouring cell will produce a non zero value. The outward flux through the right hand edge of a cell, for example, is determined to be the volume of fluid that exists in the space between the right hand edge and the width $u\delta t$, over the entire height of the cell. In the case that the cell interface passes through the cell, a scenario specific equation must be used to calculate the area of fluid that overlaps the flux area. This is illustrated with a Youngs' A-II cell in figure 14. Refer to appendix B for the full set of volume flux equations.

The flux calculation procedure produces an error for cases II and III for Orientations B and D. For the volume flux calculation, the B/D case II cells can be considered equivalent to the flux calculation of an A-III cell, as the interface goes between the top and bottom cell edges. Similarly the B/D case III cells can be treated as an A-II cell as they have an interface that goes from the left to the right cell edge. The switching of the cases at the flux calculation stage allows for the correct choice of overlap area geometry equation, which allows

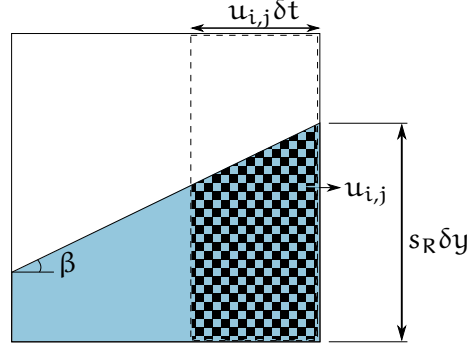


Figure 14: Volume for an example cell (Youngs' A-II). The hatched area represents the flux Fr for a positive $u_{i,j}$.
 $Fr_{i,j} = u_{i,j} \delta t (s_R \delta y - 0.5 u_{i,j} \delta t \tan(\beta))$

for the volume flux to be calculated correctly in these cases. This is reflected in algorithm 1 and can be seen more fully in the algorithms presented in appendix B.

2.2.2.1 Volume fraction advection

The volume fraction tracking method described above identifies the fluid geometry and calculates a set of volume fluxes representing the volume of fluid moving across the four cell edges. The volume fraction field for the next time step is found by applying these fluxes in an advection scheme. The current solver uses an Eulerian, operator split advection scheme with a divergence correction term, as introduced by Rider and Kothe [1998] and further discussed by Scardovelli and Zaleski [2003]. The advection equation, in vector form, with the additional divergence correction term is

$$\frac{\partial c}{\partial t} + \nabla \cdot (vc) = c \nabla \cdot v, \quad (18)$$

where v is the vector form of the velocity.

An operator split scheme requires a two stage process. Firstly a sweep of the grid is completed in one direction, locating the interface and calculating the fluxes in that direction only. These fluxes are used to advect the fluid in that direction, creating an intermediary volume fraction field c^* . The intermediary volume fractions are then used to recalculate the interface geometry and fluxes for advection in the second direction. The direction order alternates each time step to maintain stability. By splitting equation 18, discretising it and applying the fluxes as determined above, the following equations are derived.

$$c_{i,j}^* = c_{i,j}^n + \frac{Fl_{i+1,j} + Fr_{i-1,j} - Fl_{i,j} - Fr_{i,j}}{\delta x \delta y} + c_{i,j}^n \left(\frac{\delta t}{\delta x} (u_{i,j} - u_{i-1,j}) \right)$$

(19)

for advection in the x direction, and

$$c_{i,j}^{n+1} = c_{i,j}^* + \frac{Fb_{i,j+1} + Ft_{i,j-1} - Fb_{i,j} - Ft_{i,j}}{\delta x \delta y} + c_{i,j}^{n+1} \left(\frac{\delta t}{\delta x} (v_{i,j} - v_{i,j-1}) \right) \quad (20)$$

for advection in the y direction.

It must be noted that the volume fraction used in the first advection step is explicit, however the volume fraction used in the divergence correction for the second advection step is implicit; the same quantity that must be found by that equation, appears in the equation. Thus equation 20 is reformulated into an explicit form. When the x direction is the starting advection direction the procedure uses

$$c_{i,j}^* = c_{i,j}^n \left(1 + \frac{\delta t}{\delta x} (u_{i,j} - u_{i-1,j}) \right) + \frac{Fl_{i+1,j} + Fr_{i-1,j} - Fl_{i,j} - Fr_{i,j}}{\delta x \delta y}, \quad (21)$$

$$c_{i,j}^{n+1} = \frac{c_{i,j}^* + ((Fb_{i,j+1} + Ft_{i,j-1} - Fb_{i,j} - Ft_{i,j})/(\delta x \delta y))}{1 - \frac{\delta t}{\delta y} (v_{i,j} - v_{i,j-1})}. \quad (22)$$

When the y direction is the starting direction the procedure uses

$$c_{i,j}^* = c_{i,j}^n \left(1 + \frac{\delta t}{\delta y} (v_{i,j} - v_{i,j-1}) \right) + \frac{Fb_{i,j+1} + Ft_{i,j-1} - Fb_{i,j} - Ft_{i,j}}{\delta x \delta y}, \quad (23)$$

$$c_{i,j}^{n+1} = \frac{c_{i,j}^* + ((Fl_{i+1,j} + Fr_{i-1,j} - Fl_{i,j} - Fr_{i,j})/(\delta x \delta y))}{1 - \frac{\delta t}{\delta x} (u_{i,j} - u_{i-1,j})}. \quad (24)$$

2.2.3 Free Surface Boundary Conditions

The model employs the velocity and pressure boundary conditions originally partially presented by [Hirt and Shannon \[1968\]](#), the full form of these boundary conditions are given in figures 15 and 16, 17 and 18. These figures are based on the example figures presented by [Tome and McKee \[1994\]](#). It is important to note that these boundary conditions are only valid for a large curvature, as small curves may cause boundary conditions to be laid on top of on another which can cause the simulation to fail. Also the boundary conditions given in figures 17 and 18 should be applied to surface cells not contigu-

ous with any empty cells, but located diagonally adjacent to a single empty cell.

$$p - (2/Re) \left(n_x n_x \frac{\partial u}{\partial x} + n_x n_y \left(\frac{\partial y}{\partial y} + \frac{\partial v}{\partial x} + n_y n_y \frac{\partial v}{\partial y} \right) \right) = 0, \quad (25)$$

$$2n_x m_x \frac{\partial u}{\partial x} + (n_x m_y + n_y m_x) \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + 2n_y m_y \frac{\partial v}{\partial y} = 0, \quad (26)$$

where n_x and n_y are the x and y components of the outward unit normal vector and m_x and m_y are the x and y components of the tangential vector, where $m_x = n_y$ and $m_y = -n_x$. Cells adjacent to one empty cell are approximated as vertical or horizontal and equations 25 and 26 reduce to

$$p - \frac{2}{Re} \left(\frac{\partial u_n}{\partial n} \right) = 0, \quad (27)$$

$$\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = 0, \quad (28)$$

The resulting equations for cells deemed to contain a vertical or horizontal interface are given in figures 15 and 16. Cells adjacent to two empty cells or cells adjacent to no empty cells, where two of the adjacent non opposite cells are entirely filled ($C = 1$), are considered as having an interface at 45 degrees, and equations 25 and 26 reduce to

$$p = \frac{1}{Re} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) = 0, \quad (29)$$

$$\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} = 0, \quad (30)$$

The resulting equations for cells deemed to contain an interface at approximately 45° vertical or horizontal interface are given in figures 15 and 16. This is an adaptation of the method presented by Tome and McKee [1994], where only cells adjacent to two empty cells are considered as a 45° surface.

2.2.4 Flotsam and Jetsam

Flotsam and Jetsam is the term for small ejections of small quantities of fluid, shed by the main flow. These ejections can be caused by the scenario being simulated or as an unwanted by-product of the fluid tracking method. Free surface boundary conditions fail when ejections are at a scale approximately similar to the scale of the grid cells. As a result the ejections can be incorrectly assigned very high velocities. This causes a numerical slow down of the simulation, due to the required reduction in time step size to not allow the movement

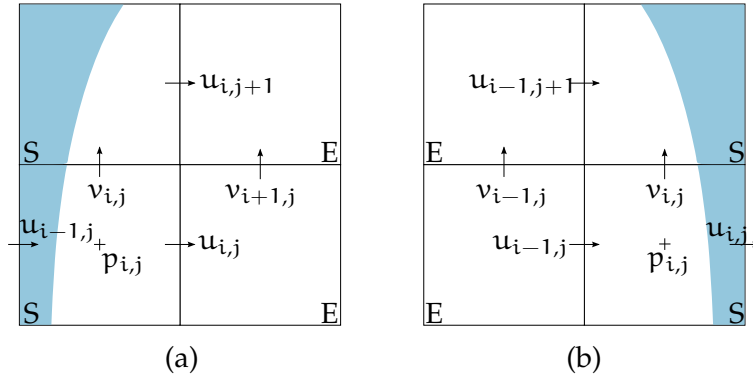


Figure 15: Vertical free surface boundary conditions, where S is a surface cell and E is an empty cell,

(a) Fluid to the left hand side, $p_{i,j} = \frac{2}{Re} \left(\frac{u_{i,j} - u_{i-1,j}}{\delta x} \right),$

$v_{i+1,j} = v_{i,j} - \frac{\delta x}{\delta y} (u_{i,j+1} - u_{i,j}),$

(b) Fluid to the right hand side, $p_{i,j} = \frac{2}{Re} \left(\frac{u_{i,j} - u_{i-1,j}}{\delta x} \right),$

$v_{i-1,j} = v_{i,j} + \frac{\delta x}{\delta y} (u_{i-1,j+1} - u_{i-1,j})$

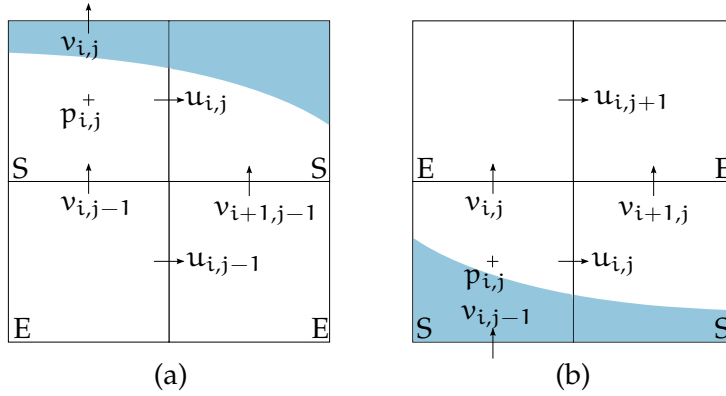


Figure 16: Horizontal free surface boundary conditions,

(a) Fluid above, $p_{i,j} = \frac{2}{Re} \left(\frac{v_{i,j} - v_{i,j-1}}{\delta y} \right),$

$u_{i,j-1} = u_{i,j} + \frac{\delta y}{\delta x} (v_{i+1,j-1} - v_{i,j-1}),$

(b) Fluid below, $p_{i,j} = \frac{2}{Re} \left(\frac{v_{i,j} - v_{i,j-1}}{\delta y} \right),$

$u_{i,j+1} = u_{i,j} - \frac{\delta y}{\delta x} (v_{i+1,j} - v_{i,j}).$

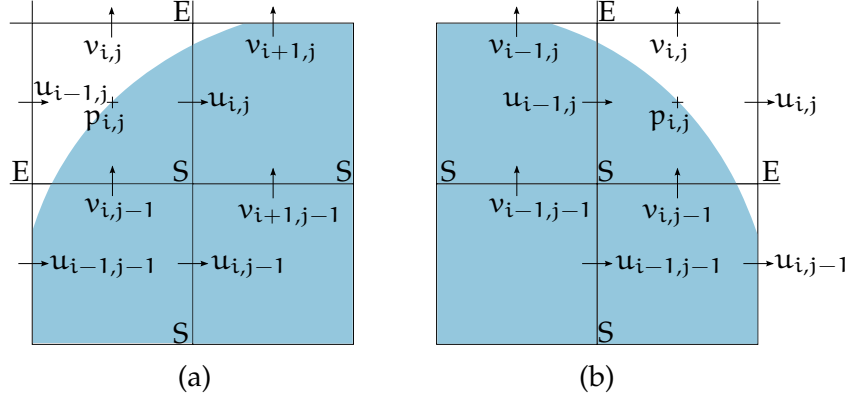


Figure 17: Corner free surface boundary conditions, (a) Fluid below and to the right hand side, $u_{i-1,j} = u_{i,j}$, $v_{i,j} = v_{i,j-1}$.

$$p_{i,j} = -\frac{1}{2\text{Re}} \frac{u_{i,j} + u_{i-1,j} - u_{i,j-1} - u_{i-1,j-1}}{\delta x} - \frac{1}{2\text{Re}} \frac{v_{i+1,j} + v_{i+1,j-1} - v_{i,j} - v_{i,j-1}}{\delta y},$$

(b) Fluid below and to the left hand side, $u_{i,j} = u_{i-1,j}$, $v_{i,j} = v_{i,j-1}$.

$$p_{i,j} = \frac{1}{2\text{Re}} \frac{u_{i,j} + u_{i-1,j} - u_{i,j-1} - u_{i-1,j-1}}{\delta x} + \frac{1}{2\text{Re}} \frac{v_{i,j} + v_{i,j-1} - v_{i-1,j} - v_{i-1,j-1}}{\delta y}$$

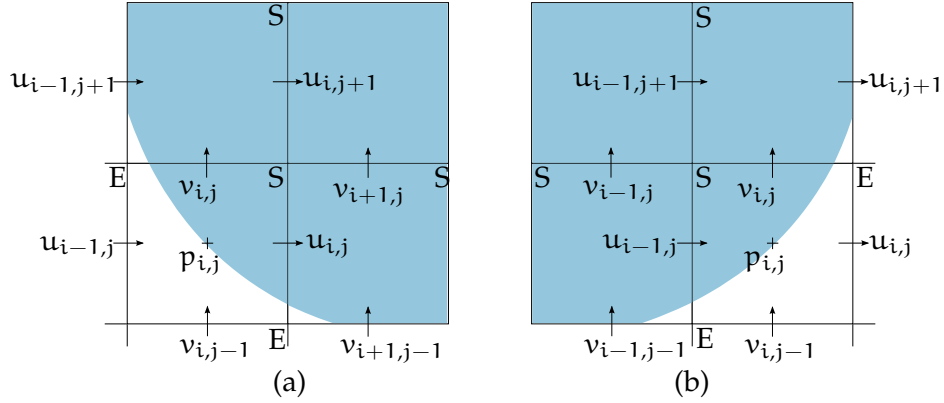


Figure 18: Corner free surface boundary conditions, (a) Fluid above and to the right hand side, $u_{i-1,j} = u_{i,j}$, $v_{i,j-1} = v_{i,j}$,

$$p_{i,j} = -\frac{1}{2\text{Re}} \frac{u_{i,j+1} + u_{i-1,j+1} - u_{i,j} - u_{i-1,j}}{\delta x} - \frac{1}{2\text{Re}} \frac{v_{i,j} + v_{i,j-1} - v_{i-1,j} - v_{i-1,j-1}}{\delta y},$$

(b) Fluid above and to the left hand side, $u_{i,j} = u_{i-1,j}$, $v_{i,j-1} = v_{i,j}$.

$$p_{i,j} = \frac{1}{2\text{Re}} \frac{u_{i,j+1} + u_{i-1,j+1} - u_{i,j} - u_{i-1,j}}{\delta x} + \frac{1}{2\text{Re}} \frac{v_{i+1,j} + v_{i+1,j-1} - v_{i,j} - v_{i,j-1}}{\delta y},$$

of the isolated fluid blob to move through more than one cell width or height.

In order to avoid the destabilizing issues caused by flotsam and jetsam, the present model searches for instances of cells which contain fluid, surrounded by empty cells. If a cell is found to match this condition, the volume of fluid is removed from the cell and added to a float that tracks volume loss in the system. The inverse scenario, where a small trapped void is surrounded by fluid, can also cause boundary condition calculation to fail, causing a slow-down and a potential overall failure of the simulation. In the case that a void was found, it could be filled in and the velocities and pressure for the cell are taken as the average of the surrounding cells. the void or ejection size is currently limited to the area of one cell.

A variation on the methods described above included scenarios where surface cells are surrounded by other surface cells on all sides. This was found to produce some improvement in numerical stability however it had the potential to significantly impact the surface profile results. The results of the use of these methods can be seen in section [4.2](#).

2.3 SEDIMENT TRANSPORT

Sediment transport is divided into bed load and suspended load. The division is made such that bed load encompasses transport by the modes of sediment sliding, rolling and saltations or small jumps. The suspended load includes all other sediment motion in the fluid bulk. In reality this division cannot be easily defined, however for the purposes of calculation, the suspended sediment transport, bed transport and transitional transport between the fluid bulk and the bed are calculated separately. Sediment transport models can be divided into those that assume local equilibrium, for bed and total (bed + suspended) load, and those that do not. A full equilibrium sediment transport model will often use an empirical formula to calculate the total transport rate and then use the Exner [1925] to calculate the bed evolution. The dynamic nature of dam break over a mobile bed means that sediment is not likely to be in equilibrium. Some improvement can be gained by applying a non equilibrium suspended sediment transport model (Advection-Diffusion in conjunction with Exner) and further improvement still can be gained by applying a fully non equilibrium total load sediment transport model, using Advection-Diffusion and a sediment transport adaptation concept. See Soares-Frazão et al. [2012] for information on a variety of current numerical models, with examples of sediment transport models that fall within each of these categories. The model presented below is a hybrid Advection-Diffusion/Exner model (modified to account for the novel Bed-Flow transition calculation), which was chosen due to the gains in capability with little to no additional computational effort.

It must be noted that the sediment transport model presented allows for the presence of turbulence despite the fact that the fluid model does not yet implement a turbulence model. As such, the implementation of the sediment model allows for either the turbulence to be neglected or for empiricisms to be used for turbulence closure until such a time when a turbulence model is implemented in the fluid simulation.

2.3.1 *Suspended Load*

The motion of sediment suspended in a fluid can be modelled by tracking the change in concentration of the sediment in each cell. The change in concentration is governed by an adapted advection-diffusion equation. A standard advection-diffusion equation allows for changes in concentration due to bulk fluid movement (advection) as well as changes due to a concentration's tendency to spread out (diffusion). A settling term is introduced to allow for the settling out of the sediment. The impact of deposition and erosion on the sus-

pendent sediment concentration is calculated separately, see 2.3.3. The operator split, adapted advection-diffusion equation for suspended sediment transport is

$$\frac{\partial C'}{\partial t'} + u' \frac{\partial C'}{\partial x'} - \frac{\partial}{\partial x'} \left(K'_x \frac{\partial C'}{\partial x'} \right) = 0, \quad (31)$$

$$\frac{\partial C'}{\partial t'} + v' \frac{\partial C'}{\partial y'} + v'_s \frac{\partial C'}{\partial y'} - \frac{\partial}{\partial y'} \left(K'_y \frac{\partial C'}{\partial y'} \right) = 0, \quad (32)$$

where C' is the sediment concentration, v'_s is the settling velocity, K'_x is the longitudinal coefficient of diffusion, and K'_y is the vertical coefficient of diffusion. Refer to section 2.3.5 for further details on the calculation of the coefficients of diffusion.

The following ND variables are defined in order to non-dimensionalise the operator split advection-diffusion equations for suspended sediment transport.

$$\begin{aligned} C &= \frac{C'}{\rho'_{\text{pack}}} & t &= \frac{t'U'}{L'} \\ x &= \frac{x'}{L'} & y &= \frac{y'}{L'} \\ u &= \frac{u'}{U'} & v &= \frac{v'}{U'} \\ K_x &= \frac{K'_x}{L'U'} & K_y &= \frac{K'_y}{L'U'} \end{aligned}$$

where C is the ND sediment concentration, ρ'_{pack} is the at rest concentration of sediment particles in the bed, K_x is the ND horizontal diffusion coefficient and K_y is the ND vertical diffusion coefficient. The value of ρ_{pack} is equal to the density of the sediment material ρ_s multiplied by the packing density, where the packing density is the fraction of a volume filled by the sediment when packed¹. Thus the ND advection-diffusion equation is:

$$\frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} - \left[\frac{\partial}{\partial x} \left(K_x \frac{\partial C}{\partial x} \right) \right] = 0 \quad (33)$$

$$\frac{\partial C}{\partial t} + v \frac{\partial C}{\partial y} + v_s \frac{\partial C}{\partial y} - \left[\frac{\partial}{\partial y} \left(K_y \frac{\partial C}{\partial y} \right) \right] = 0 \quad (34)$$

¹ Mathematical calculation of particle packing has been of interest for over 400 years. The Kepler conjecture [Kepler, 1611] states that either cubic or hexagonal close packing (with a packing density of $\pi/(3\sqrt{2}) \approx 74\%$) is the densest possible configuration of equally sized spheres in three dimensions. A proof of this conjecture was completed in 1998 by Hales, however verification proved difficult and was only published in 2005 [Hales, 2005]. A formal proof (including a computerised procedure for self verification) was presented by Hales et al. in 2015. Natural beds will not be ordered in the specific pattern for maximum packing. Random close packing of spheres achieves an approximate packing density of approximately 64% [Jaeger et al., 1992]. Particles with other shapes can have larger packing densities (e.g. the maximum packing density and the average random packing density of ellipsoids is greater than those of spheres Donev et al. [2004])

This equation was discretised with a backward finite difference approximation so as to calculate the sediment concentration field after the time step δt . In addition, the volume fluxes F_l , F_r , F_t and F_b that were previously calculated for the free surface volume tracking calculation can be substituted into the equation in place of the advection component. The volume fluxes are defined as positive volumes of fluid moving outward from the indexed cell through the subscripted cell edge, where r is the right edge, l is the left edge, t is the top edge and b is the bottom edge. Thus, to calculate sediment flux we can simply multiply the volume flux by the concentration of sediment in the cell from which the volume flux originated. Note that the boundary condition for the advection component is maintained by setting the volume fluxes on the boundary (e.g. $F_L = 0$ at a rigid left hand boundary). The discretised form of equation 33 is

$$C^{n+1} = C^n + \frac{F_{l+1,j}C_{i+1,j} + F_{r-1,j}C_{i-1,j} - F_{l,j}C_{i,j} - F_{r,j}C_{i,j}}{\delta x \delta y} + \frac{\delta t}{\delta x} \left(\frac{Kx_i(C_{i+1,j} - C_{i,j})}{\delta x} - \frac{Kx_{i-1}(C_{i,j} - C_{i-1,j})}{\delta x} \right). \quad (35)$$

An adaptation to equation 35 is required for cells that are boundary adjacent so that material is not diffused across the boundary. For cells adjacent to a boundary on the right, the following equation is employed:

$$C^{n+1} = C^n + \left(\frac{F_{l+1,j}C_{i+1,j} + F_{r-1,j}C_{i-1,j} - F_{l,j}C_{i,j} - F_{r,j}C_{i,j}}{\delta x \delta y} + \frac{\delta t}{\delta x} \left(-\frac{Kx_{i-1}(C_{i,j} - C_{i-1,j})}{\delta x} \right) \right) \quad (36)$$

For cells adjacent to a boundary on the left, the following equation is employed:

$$C^{n+1} = C^n + \left(\frac{F_{l+1,j}C_{i+1,j} + F_{r-1,j}C_{i-1,j} - F_{l,j}C_{i,j} - F_{r,j}C_{i,j}}{\delta x \delta y} + \frac{\delta t}{\delta x} \left(\frac{Kx_i(C_{i+1,j} - C_{i,j})}{\delta x} \right) \right) \quad (37)$$

If the cell in question contains the bed surface then a further adaptation is required to account for the reduced viable cell size. Equation 38 is the equation for a bed cell that does not border a boundary. Equations for the bed cells that do contact the left and right boundary can be found by similarly multiplying the components by the factor

$\delta y/(\delta y - r_b)$, where r_b is the bed height within the final/highest bed cell, for a given width.

$$C^{n+1} = C^n + \frac{\delta y}{\delta y - r_b} \left(\frac{Fl_{i+1,j}C_{i+1,j} + Fr_{i-1,j}C_{i-1,j} - Fl_{i,j}C_{i,j} - Fr_{i,j}C_{i,j}}{\delta x \delta y} + \frac{\delta t}{\delta x} \left(\frac{Kx_i(C_{i+1,j} - C_{i,j})}{\delta x} - \frac{Kx'_{i-1}(C_{i,j} - C_{i-1,j})}{\delta x} \right) \right) \quad (38)$$

The finite difference approximation for the y-direction is as follows

$$C^{n+1} = C^n + \frac{Fb_{i,j+1}C_{i,j+1} + Ft_{i,j-1}C_{i,j-1} - Fb_{i,j}C_{i,j} - Ft_{i,j}C_{i,j}}{\delta x \delta y} + \delta t \left[\frac{v'_s}{U'} \left(\frac{C_{i,j+1} - C_{i,j}}{\delta y} \right) + \frac{1}{\delta y} \left(\frac{Ky_{i,j}(C_{i,j+1} - C_{i,j})}{\delta y} - \frac{Ky_{i,j-1}(C_{i,j} - C_{i,j-1})}{\delta y} \right) \right] \quad (39)$$

Once again an adaptation is required for cells adjacent to boundaries, in this case to not allow diffusion or settling to occur across the boundary. Note that material that settles out is accounted for separately in the deposition calculation, see 2.3.3. For cells adjacent to an upper boundary, the following equation is employed:

$$C^{n+1} = C^n + \frac{Fb_{i,j+1}C_{i,j+1} + Ft_{i,j-1}C_{i,j-1} - Fb_{i,j}C_{i,j} - Ft_{i,j}C_{i,j}}{\delta x \delta y} + \delta t \left[\frac{v'_s}{U} \left(\frac{-C_{i,j}}{\delta y} \right) + \frac{1}{\delta y} \left(-\frac{Ky_{i,j-1}(C_{i,j} - C_{i,j-1})}{\delta y} \right) \right] \quad (40)$$

For cells adjacent to an lower boundary, the following equation is employed:

$$C^{n+1} = C^n + \frac{Fb_{i,j+1}C_{i,j+1} + Ft_{i,j-1}C_{i,j-1} - Fb_{i,j}C_{i,j} - Ft_{i,j}C_{i,j}}{\delta x \delta y} + \delta t \left[\frac{v'_s}{U} \left(\frac{C_{i,j+1}}{\delta y} \right) + \frac{1}{\delta y} \left(\frac{Ky_{i,j}(C_{i,j+1} - C_{i,j})}{\delta y} \right) \right] \quad (41)$$

If the cell in question contains the bed surface, the equation must again be adapted such that the reduced viable cell sized is accounted for, as well as not allowing any material to be lost through the settling

or diffusion components, as with the case of cell adjacent to the lower boundary.

$$C^{n+1} = C^n + \frac{\delta y}{\delta y - r_b} \left(\frac{Fb_{i,j+1}C_{i,j+1} + Ft_{i,j-1}C_{i,j-1} - Fb_{i,j}C_{i,j} - Ft_{i,j}C_{i,j}}{\delta x \delta y} + \delta t \left[\frac{v'_s}{U} \left(\frac{C_{i,j+1}}{\delta y} \right) + \frac{1}{\delta y} \left(\frac{Ky_{i,j}(C_{i,j+1} - C_{i,j})}{\delta y} \right) \right] \right) \quad (42)$$

2.3.2 Bed Load

The evolution of the bed surface height due to bed sediment transport is modelled here by the Exner equation [Exner, 1925]. The standard form of the Exner equation in one dimension is

$$\frac{dh'}{dt'} = -\frac{1}{\rho'_{pack}} \frac{dq'_s}{dx'} \quad (43)$$

where h' is bed height, and q'_s is the sediment mass flux due to bed load. Refer to section 2.3.5 for further details on the calculation of the volumetric bed sediment flux. By adding in a sink and source term to allow for deposition and erosion and substituting in the volumetric sediment flux, $q'_{s,v}$, in the place of the sediment mass flux, where $q'_{s,v} = q'_s/\rho_{pack}$, the following equation is derived.

$$\frac{dh'}{dt'} = \frac{(Q' - S')}{\rho_{pack}} + \frac{dq'_{s,v}}{dx'}, \quad (44)$$

where Q' in two dimensions is the mass deposited per width, per time and S' is the mass eroded per width, per time. Using the following ND variables, and the definition for the mass deposition and erosion rates presented by Soltanpour and Jazayeri [2009], a non-dimensionalised Exner equation can be derived.

$$\begin{aligned} Q &= \frac{Q'}{\rho'_{pack}U'} = \frac{v'_s}{U'} C \left(1 - \frac{\tau}{\tau_D}\right) \delta x, \\ S &= \frac{S'}{\rho'_{pack}U'} = E|u_*| \left(\frac{\tau}{\tau_D} - 1\right), \\ q_{s,v} &= \frac{q'_{s,v}}{L'U'}, \\ h &= \frac{h'}{L'}, \end{aligned}$$

where Q is the ND sediment mass deposition rate, S is the ND sediment mass erosion rate, τ is the bed shear stress, τ_D is the critical

shear threshold for deposition, u_* is the **ND** bed shear velocity, $q_{s,v}$ is the **ND** volumetric bed load, E is a Coefficient of Erosion (chosen to be equal to 1 with no other information), h' is the bed height and h is the **ND** bed height. By discretising the differential terms (using a forward difference for time and a backward difference for the spatial term), the following equations are derived.

$$h_{n+1} = h_n + dt(Q - S) - \frac{q_{s,v_i} - q_{s,v_{i-1}}}{\delta x}. \quad (45)$$

The above equation is correct so long as, for depositional cases, the area from which material is taken (the sediment flux area) is less than the remaining viable area of the cell. In the case that the flux area is greater than the viable cell area, material from the cell above is taken for deposition, which may have a different concentration to the primary bed cell. As an alternative to the calculation of the values Q and S , it is possible to calculate the depositional or erosional sediment flux height F_H , for a specific cell width δx .

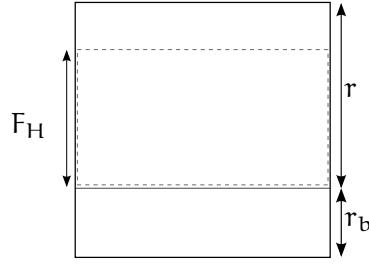


Figure 19: Illustration of the flux height, and the bed cell remainder terms.

$$F_H = \begin{cases} \frac{v_s}{U} (1 - \frac{\tau}{\tau_D}) \delta t, & \text{when } \tau < \tau_D \\ E|u_*|(\frac{\tau}{\tau_D} - 1) \delta t, & \text{when } \tau > \tau_D \end{cases} \quad (46)$$

In the depositional case where F_H is also greater than the remaining viable height of the cell, r , the height gain equivalent of the mass deposited, H^* , will be equal to

$$H^* = (F_H - r)C_{j+1} + rC_j \quad (47)$$

In the case of deposition and F_H is less than r then the height gain due to deposition is simply

$$H^* = F_H C_j \quad (48)$$

In the case of erosion, whether or not the erosive sediment flux height drops below bottom edge of the cell, the height loss due to erosion is simply

$$H^* = -F_H \quad (49)$$

as the concentration in the bed in all cells is said to be equal to the packing density and so the [ND](#) concentration is equal to 1 in the entire bed. In an effort to make the equations consistent, a variable h_{qs} is defined to represent the height change due to bed load. Using these terms it is possible to repose the Exner equation as a summation of height changes, see equation [51](#).

$$h_{qs} = -\frac{q_{s,v_i} - q_{s,v_{i-1}}}{\delta x} \delta t \quad (50)$$

$$h_i^{n+1} = h_i^n + H^* + h_{qs} \quad (51)$$

2.3.3 Fluid-Bed Transition

In order to properly conserve sediment mass at the bed surface, a transitional model is applied. The amount that a cell's height will increase or decrease must properly impact the scalar representation of the concentration within the cells that are near them. To correctly calculate the amount of material to be deposited or eroded in a time step, for given specific width, the flux area height must be found. When sediment flux is calculated it can be thought of as equal to the area that will transition multiplied by the concentration of sediment within this area. The width of the area is the cell width δx , however the height must be calculated. Deposition or erosion is chosen, depending on the shear stress conditions at the bed surface, and the flux area height F_H is calculated. From this value, the height equivalent of the total mass deposited or eroded, H^* , is calculated.

To calculate the change in suspended sediment concentration, due to both the change in mass in the cell and the change in fluid volume (area in 2D), the scenario under which it falls can be found.

Primarily the scenarios can be sorted into three cases: depositional ($H^* > 0$), erosional ($H^* < 0$) or neither depositional nor erosional ($H^* = 0$). The third case is necessary for the case where the area of the cell is changed by bed load, causing a change in the concentration independent to the introduction or removal of mass from the cell. Figure [20](#) illustrates all of the fluid-bed transition scenarios, grouped into the three cases, *a-e*, *f-i* and *j-l*.

In the depositional case there are five distinct scenarios. In scenario *a* the flux area height breaches the cell above however the height remains in the cell or decreases into the cell below. In this case, all of the sediment in the primary cell is deposited leaving it with a concentration of zero, however the cell above also loses mass, without losing area.

$$C_j = 0$$

$$C_{j+1} = \frac{C_{j+1} \delta x \delta y - (H^* - h^*) \delta x}{\delta x \delta y} \quad (52)$$

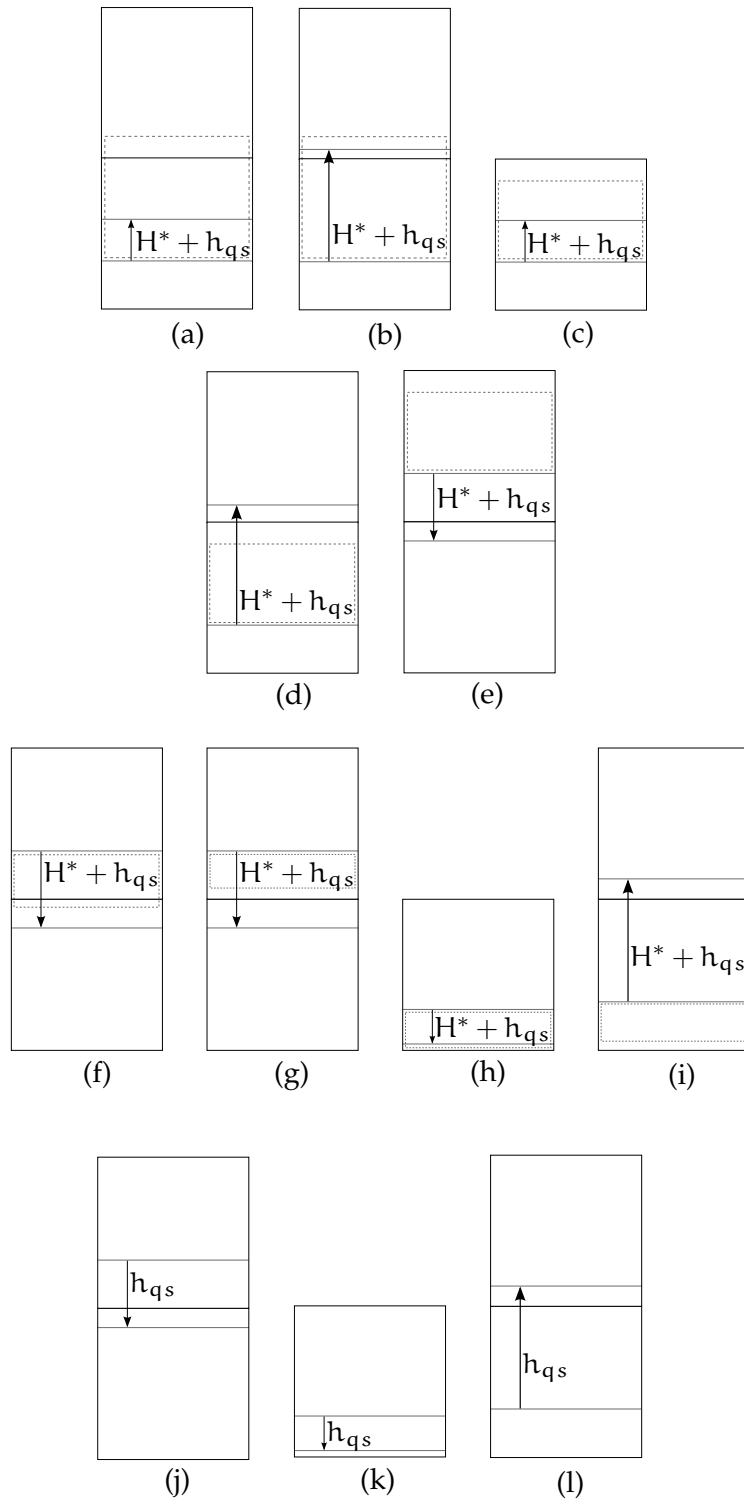


Figure 20: Fluid-bed transition scenarios. Scenarios (a)–(e) represent deposition, (f)–(i) represent erosion and (j)–(l) represent changes in bed height due solely to bed load with no erosion or deposition. Scenarios (c), (h) and (k) are shown with a specific bed height change, however these cases apply to bed height changes in either direction, where the bed height remains in the cell.

for $H^* > 0$, $H^* > h^*$, $H^* + h_{qs} \leq r$, where h^* is the height equivalent of the suspended sediment of the primary cell if that sediment was in the bed, which is equal to $h^* = C_j r$.

Scenario *b* also sees the flux area height breach the cell above with the new bed height also reaching the cell above. Again the primary cell has no suspended sediment concentration (now being part of the bed), and the cell above loses material whilst also having a reduction in cell area.

$$C_j = 0$$

$$C_{j+1} = \frac{C_{j+1} \delta x \delta y - (H^* - h^*) \delta x}{\delta x (\delta y - (H^* + h_{qs} - r))} \quad (53)$$

for $H^* > 0$, $H^* > h^*$, $H^* + h_{qs} > r$.

In scenario *c* the flux height area does not breach the cell above, and the height remains within this cell.

$$C_j = \frac{C_j r \delta x - H^* \delta x}{\delta x (\delta y - (H^* + h_{qs} + r_b))} \quad (54)$$

for $H^* > 0$, $H^* \leq h^*$, $r_b \leq H^* + h_{qs} \leq r$.

In scenario *d* the flux height area again does not breach the cell above, however the height does increase into the cell above.

$$C_{j+1} = \frac{C_{j+1} \delta x \delta y + (H^* - r) \delta x}{\delta x (\delta y - (H^* + h_{qs} - r))} \quad (55)$$

for $H^* > 0$, $H^* \leq h^*$, $H^* + h_{qs} > r$.

In scenario *e* the height of the bed drops into the cell beneath however the depositional flux height remains inside the cell so some material may remain in this cell, whose area is now a full cell area.

$$C_j = \frac{C_j r \delta x - H^* \delta x}{\delta x \delta y} \quad (56)$$

for $H^* > 0$, $H^* + h_{qs} < 0$, $|H^* + h_{qs}| > r_b$.

It might be assumed that there would be a matching case for when the height drops into the cell below, and the depositional flux height breaches the cell above. This is unnecessary as in this case all of the material from the original primary cell has been deposited, and so the concentration in that cell is zero. Thus the equations are identical to those given for scenario *a*.

In the case of erosion, there are four distinct scenarios. The first two of these scenarios, *f* and *g*, are for the case where the new bed height decreases into the cell beneath. In scenario *f*, the height of material eroded is greater than the bed height remainder left in the cell, thus material is eroded from the cell below.

$$C_j = \frac{C_j \delta x r + r_b \delta x}{\delta x \delta y}$$

$$C_{j-1} = \frac{(-H^* - r_b) \delta x}{\delta x (-(r_b + H^* + h_{qs}))} \quad (57)$$

for $H^* < 0$, $H^* + h_{qs} < 0$, $|H^* + h_{qs}| > r_b$, $|H^*| > r_b$. Note that the sediment is not summed and spread across the entire viable area. Instead the material is added to the concentration of the cell from which the material originated.

In Scenario *g*, the erosional height is less than the bed height remainder, thus the bed height movement into the cell below is due to bed load.

$$C_j = \frac{C_j \delta x r + H^* \delta x}{\delta x \delta y}$$

$$C_{j-1} = 0 \quad (58)$$

for $H^* < 0$, $H^* + h_{qs} < 0$, $|H^* + h_{qs}| > r_b$, $|H^*| \leq r_b$.

The third, scenario *h*, is where the new bed height remains within the primary cell.

$$C_j = \frac{C_j \delta x r - H^* \delta x}{\delta x (\delta y - (r_b + H^* + h_{qs}))} \quad (59)$$

for $H^* < 0$, $H^* + h_{qs} < 0$ and $|H^* + h_{qs}| \leq r_b$ or $H^* + h_{qs} > 0$ and $H^* + h_{qs} \leq r$.

The fourth, scenario *i*, is where the bed height increases into the cell above.

$$C_{j+1} = \frac{C_{j+1} \delta x \delta y - (H^* - h^*) \delta x}{\delta x (\delta y - (H^* + h_{qs} - r))} \quad (60)$$

for $H^* < 0$, $H^* + h_{qs} > 0$, $H^* + h_{qs} > r$.

The final case occurs when there is a height change due to bed load with no deposition or erosion. There are three scenarios. The first, scenario *j*, is where the bed height decreases into the cell beneath.

$$C_j = \frac{C_j \delta x r}{\delta x \delta y} \quad (61)$$

for $H^* = 0$, $h_{qs} < 0$, $|h_{qs}| > r_b$.

The second, scenario *k*, is where the new bed height remains within the primary cell.

$$C_j = \frac{C_j \delta x r}{\delta x (\delta y - (h_{qs} + r_b))} \quad (62)$$

for $H^* = 0$, $h_{qs} < 0$ and $|h_{qs}| \leq r_b$ or $h_{qs} > 0$ and $h_{qs} \leq r$.

The third, scenario *l*, is where the new bed height increases into the above cell.

$$C_{j+1} = \frac{C_{j+1} \delta x \delta y + h^* \delta x}{\delta x (\delta y - (h_{qs} - r))} \quad (63)$$

for $H^* = 0$, $h_{qs} > 0$, $h_{qs} > r_b$.

2.3.4 Recursive redistribution of suspended sediment concentration

It is possible for the suspended sediment in the cell to increase beyond the maximum value of 1 (equivalent to the packing density of that sediment). In this case a recursive algorithm can be applied to reallocate the additional sediment. The algorithm is called with the location indices, (i, j) of the overshoot cell and that cell's viable cell height r . Using these values, the algorithm calculates the vertical index j^{it} of the next cell above the primary cell to have a concentration less than 1. The additional material is added to this cell using the following equation

$$C_{j^{it}} = \frac{C_{j^{it}}\delta x\delta y + (c_j - 1)\delta x r}{\delta x\delta y}$$

$$C_j = 1$$

Following this recalculation, if the concentration of the upper cell is now greater than 1, the algorithm recursively calls itself using the location indices of the upper cell and the remainder (which equals a full cell height δy). This algorithm will continue to spread any overshoot material upwards until no cells exist with a concentration greater than 1. Both the fluid-bed transition and the overshoot redistribution algorithms deal only with vertical movement. Lateral spreading of material is accounted for in the bed by the bed load calculation. In the flow, lateral spread is accounted for by advection and lateral diffusion.

2.3.5 Empirical Relationships

Empirical relationships are used for the calculation of the bed load, the bed shear stress and the longitudinal and vertical coefficients of diffusion.

The equation given by Meyer-Peter and Müller [1948] is used to calculate the volumetric bed load for each width of the grid. The equation is as follows:

$$q_{s,v} = \sqrt{(s-1)gd_s^3} \left(\frac{4\tau}{\rho(s-1)gd_s} - \tau_c \right)^{\frac{3}{2}} \quad (64)$$

where $q_{s,v}$ is the volumetric bed load, s is the specific density, where $s = \rho_s/\rho$, d_s is the sediment particle diameter, τ is the bed shear stress, where $\tau = \rho u_*^2$ and τ_c is the [ND](#) critical shear stress indicating the initiation of motion. The Meyer-Peter and Müller equation was also employed in the recent coupled Navier-Stokes-Exner model presented by Burkow and Griebel [2013], as well as in seven of the twelve models compared by Soares-Frazão et al. [2012].

To calculate the bed shear stress, the shear velocity u_* is estimated by using the “law of the wall”, where the velocity is said to be pro-

portional to the natural logarithm of the distance away from the wall in question. The equation for this is

$$u = \frac{u_*}{\kappa} \ln \left(\frac{z}{z_0} \right) \quad (65)$$

where κ is the von Karman constant, z is the distance from the wall (in this case the bed) and z_0 is the friction length. By substituting in the friction length as determined empirically by Wilcock [1996] and reorganising the equation, the following equation for the shear velocity can be reached,

$$u_* = \frac{u\kappa}{\ln \left(\frac{z}{\alpha d_p / 30} \right)} \quad (66)$$

where α is an empirical factor that equals $\alpha = 2.85$ and d_p is the particle diameter size for which a proportion p is less than.

The present model uses the Van Rijn [1984a] formulation of the Shields diagram [Shields, 1936] to calculate the critical shear stress.

$$\tau_c = \begin{cases} 0.24d_s^{*-1} & d_s^* \leq 4 \\ 0.14d_s^{*-0.64} & 4 < d_s^* \leq 10 \\ 0.04d_s^{*-0.1} & 10 < d_s^* \leq 20 \\ 0.013d_s^{*0.29} & 20 < d_s^* \leq 150 \\ 0.14d_s^{*-0.64} & d_s^* > 150 \end{cases} \quad (67)$$

where d_s^* is a dimensionless sediment particle size parameter equal to $d \left(\frac{(s-1)g}{\nu^2} \right)^{\frac{1}{3}}$. The shields critical shear stress calculation is an empirical relationship based on results gained from the experimental study of different steady state discharges over sediment beds. The application of this relationship to an unsteady model may increase the stress associated with initiation of motion, causing a reduced approximation to both the bed load and the amount of erosion.

The empirical study of diffusion and dispersion in the vertical, lateral and longitudinal directions have produced many empirical relationships. Firstly the coefficient of diffusion in each of the directions cannot be expected to be the same. Diffusion by turbulent eddy mixing is often the primary cause in the vertical direction, where bulk mixing dispersion may be the primary cause in the longitudinal di-

rection. The present model calculates a vertical coefficient of diffusion via the equations given by [Van Rijn \[1984b\]](#),

$$\beta = 1 + C_A \left[\frac{v_s}{u_*} \right]^{C_B} \quad (68)$$

$$\epsilon_{f,\max} = 0.25\kappa u_* \quad \text{for } \frac{z}{d_f} \geq 0.5 \quad (69)$$

$$\epsilon_f = 4 \frac{z}{d_f} \left(1 - \frac{z}{d_f} \right) \epsilon_{f,\max} \quad \text{for } \frac{z}{d_f} < 0.5 \quad (70)$$

$$\phi = 1 + C^{0.8} - 2C^{0.4} \quad (71)$$

$$Ky = \beta \phi \epsilon_f \quad (72)$$

where ϵ_f is the coefficient of fluid momentum or kinematic eddy viscosity, $\epsilon_{f,\max}$ is the maximum coefficient of fluid momentum, β is the factor that relates the diffusion of a fluid “particle” to the diffusion of a discrete sediment particle and ϕ is a factor included to account for the damping effect of the concentration on the fluid turbulence. [Van Rijn](#) defines empirical factors $C_A = 2$ and $C_B = 2$, however [Kerssens et al. \[1979\]](#) found $C_A = 1.54$ and $C_B = 2.12$ to produce a better fit. The damping parameter ϕ is used by various authors as an aid to fitting the model to experimental data.

The horizontal or longitudinal coefficient of diffusion is calculated using Elder’s equation [[Elder, 1959](#)] for an infinitely wide channel.

$$Kx = 5.9d_f u_* \quad (73)$$

where d_f is the total fluid depth for the given width. This equation assumes the flow is uniform, an assumption that isn’t maintained by the present model, however it is useful as a conservative approximation.

The model, at present allows for the settling velocity as an input variable, with the sediment density, particle size and packing density. However, in the case that a simulation is to be run without knowledge of the settling velocity the present model allows for an approximation to be calculated using the procedure presented by [Van Rijn \[1984b\]](#). In addition to the calculation of the settling velocity, Van Rijn discusses the impact of the presence of other particles in suspension on the settling velocity of the particles, noting that increasing concentration reduces the settling velocity of those particles, a phenomenon referred to as hindered settling. The equations for calculation of the

settling velocity and the hindered settling velocity reduction factor are

$$v_s = \frac{(s-1)gd^2}{18\nu} \quad 1 < d \leq 100\mu\text{m} \quad (74)$$

$$v_s = \frac{10\nu}{d} \left[\left(1 + \frac{0.01(s-1)gd^3}{\nu^2} \right)^{0.5} - 1 \right] \quad 100 < d < 1000\mu\text{m} \quad (75)$$

$$v_s = 1.1[(s-1)gd]^{0.5} \quad d \geq 1000\mu\text{m} \quad (76)$$

$$v_{s,m} = (1 - 2.15C)(1 - 0.75c^{0.33})v_s \quad (77)$$

where $v_{s,m}$ is the particle fall velocity in a suspension, and v_s is the particle fall velocity in clear fluid. Note that equation 74 is the Stokes equation for the movement of a sphere falling through a liquid [Stokes, 1851]. For an alternative procedure for the calculation of the settling velocity of irregular particles, refer to the empirical relationship produced by Dietrich [1982] based on the particle Reynolds number.

2.3.6 Critical Angle of Repose Redistribution

The critical angle of repose redistribution algorithm is a method for the redistribution of the discrete heights of a modelled bed, such that the angle of the bed slope remains below a critical value. Initially a local smoother was enacted, which calculated the local gradient or the gradient between neighbouring discrete bed heights for every column in the grid. If the gradient was greater than the critical angle of repose, the algorithm would artificially raise and lower the bed heights at the point in question such that the slope angle was equal to the angle of repose. Local gradient correction caused gradients either side of the point to increase, potentially beyond the critical gradient. Local gradient correction required a large number of repetitions and caused a slow-down of the simulation as a whole. In the real world, sediment that exists as part of a slope with an angle greater than the critical angle of repose will be unstable and will shift down the slope. In the case of a heap of material on a flat base, natural redistribution will cause sediment to move downwards and outward, reducing the height of the peak and increasing the width of the slope. The aim of a redistribution algorithm is to mimic this natural process to model bed morphology within the confines of a slope limit.

A more efficient algorithm was developed that studied the gradients on a larger scale, between peaks and troughs. Peaks are defined as locations where, when moving from left to right across the domain, the gradients go from being positive to negative ($h_i - h_{i-1}/\delta x > 0$

and $h_{i+1} - h_i/\delta x < 0$). Troughs are defined as locations where, when moving from left to right across the domain, the gradients go from being negative to positive ($h_i - h_{i-1}/\delta x < 0$ and $h_{i+1} - h_i/\delta x > 0$). Peaks and troughs are found using this definition and their locations are marked and stored as the vector PTL (Peak/Trough Locations).

Carrigy [1970] defines an upper and lower angle of repose, where the upper or critical angle of repose α_c is the maximum angle a slope can reach before failing and the lower angle or angle of rest, α_r is the angle of the slope once avalanching has ceased. The value of the critical angle of repose is dependant on many factors (e.g. the medium, the volume of the particles, their specific gravity and their shape). Typically the values for a material's critical angle of repose will lie between 25° and 40° [Carrigy, 1970], with natural sands having an angle around 35° . The angle of rest is typically quite close to the critical angle (within $2-5^\circ$) and so, for simplicity, the procedures described below determine if a slope has an inclination greater than the critical angle and, if required, redistributes it down to the critical angle. An angle of rest can easily be incorporated into future work by separating the angle used for slope determination and the angle used to set newly redistributed slopes.

Preliminary testing of the algorithm revealed that additional peak/trough location markers needed to be placed to mark the locations of plateaus, where plateaus are defined as cells where the heights from one cell to the next are identical. Plateaus must be marked at their start and end point, as the slope exists between the plateau start or end and the neighbouring marked peak/trough location. Marking the plateau at the centre point would increase the base width of the slope, where the base width is the horizontal distance between markers. Increasing the width of the base reduces the perceived slope angle and as such a slope with an angle slightly greater than the critical angle will be perceived as having a slope with an angle less than the critical angle, and no redistribution will occur. The value of the plateau length (PL) is separately recorded as the length between the plateau start and end markers.

Location markers were also required for jumps, where a jump is defined as a sharp increase in height, greater than a given value, whilst the sign of the gradient remains unchanged. Initially this value was set to a single cell's height, δy , however the critical local gradient this would enforce would be dependant on the ratio of δx to δy . Thus this value is set to δx as this enforces a known local limit of 45° . A jump located within a slope does not require an additional peak/trough location marker, however a boolean jump marker, J_b , must be placed to indicate the existence of the jump between the two location markers and to force redistribution no matter the apparent gradient. In cases where jumps are located before or after plateaus, peak/trough location markers may need to be placed. The proce-

cedure for the calculation of peak/trough locations (PTL), boolean jump markers (J_b) and plateau lengths (PL) is given in algorithm 4. Each of the bed conditions are illustrated in figure 21. Once the peak, trough, plateau and jump locations are found, the gradient between locations can be calculated and compared to the critical gradient. If the gradient of the slope is greater than the critical gradient, the slope must undergo redistribution or smoothing.

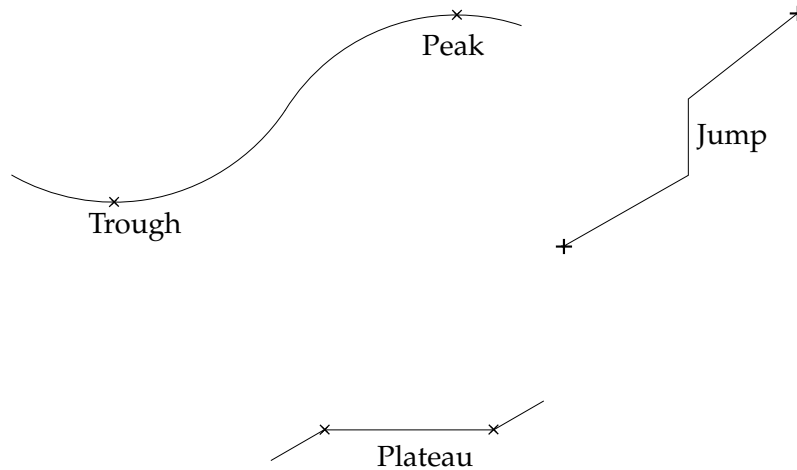


Figure 21: Peaks, troughs, jumps and plateaus

The smoothing of peaks and troughs must be considered separately. Consider a small triangular peak of sediment on a wide flat base. If the slope angle is greater than the critical angle of repose, the sediment will collapse such that the height of the triangle decreases and the base length of the triangle increases. Thus a smoothing algorithm must smooth peaks outwards from the centreline of a peak. Consider a small triangular trough in an otherwise flat wide sediment bed. If the slope angle is greater than the critical angle of repose the sediment surrounding the trough will collapse in, filling in the trough. This is an increase in bed height at the trough centreline and an increase in base length of the triangular void. Thus the algorithm must smooth troughs by smoothing the void area outwards from the centreline of the trough.

The previously discussed peak trough markers are insufficient to properly determine whether a given slope should be smoothed as a peak or a trough. Incorrectly smoothing a trough as a peak will often cause unrealistic increases in bed height, see figure 22. Incorrectly smoothing a peak as a trough can cause unrealistic decreases in bed height.

In order to judge when to smooth a slope as a peak and when to smooth a slope as a trough, a “peakiness” or P-factor (F_P) and a

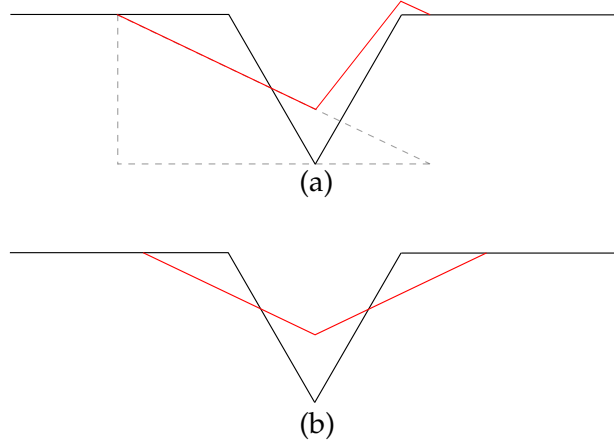


Figure 22: Illustration of the consequence of smoothing a trough feature using the peak smoothing method. Figure (a) shows the original trough in black, redistribution via the peak method in red and the calculated new triangle that led to the redistribution in dashed grey. Figure (b) shows the correct redistribution of a trough using the trough method.

“troughiness” or T-factor (F_T) is defined to judge the areas of sediment either side of a peak or trough point respectively. For the T-factor, if the area of sediment between a trough point and the previous peak/trough point divided by the area of sediment between the trough point and the next peak/trough point is between 0.25 and 4 the trough is considered to be the prominent feature and the slope should be smoothed as a trough. When the factor lies outside this range the prominent feature is considered to be the peak and should be smoothed as such. The peakiness factor is similarly calculated for the areas either side of a peak. The P-factor indicates a prominent peak within the limits of 0.25 and 4 and indicates a trough outside of those limits. The value of the limits is derived from the point at which the area of the trough void below a mid line is equal to the area of sediment above the mid line in an adjacent peak. An illustration of the peaks and troughs and where they would lie within the P-factor and T-factor ranges is shown in figure 23. The equations for the determination of the P-factor are

$$\text{sum}_{L,p} = \sum_{i=\text{PTL}[a-1]}^{i=\text{PTL}[a]} (h[i] - h[\text{PTL}[a-1]]), \quad (78)$$

$$\text{sum}_{R,p} = \sum_{i=\text{PTL}[a]}^{i=\text{PTL}[a+1]} (h[i] - h[\text{PTL}[a+1]]), \quad (79)$$

$$F_p = \frac{\text{sum}_{L,p}}{\text{sum}_{R,p}}, \quad (80)$$

where $\text{sum}_{L,p}$ is the sum of the sediment heights left of a peak, $\text{sum}_{R,p}$ is the sum of the sediment heights right of a peak and $\text{PTL}[a]$ is the current “Peak/Trough location”. The equations for the determination of the T-factor are

$$\text{sum}_{L,t} = \sum_{i=\text{PTL}[a-1]}^{i=\text{PTL}[a]} (h[i] - h[\text{PTL}[a]]), \quad (81)$$

$$\text{sum}_{R,t} = \sum_{i=\text{PTL}[a]}^{i=\text{PTL}[a+1]} (h[i] - h[\text{PTL}[a]]), \quad (82)$$

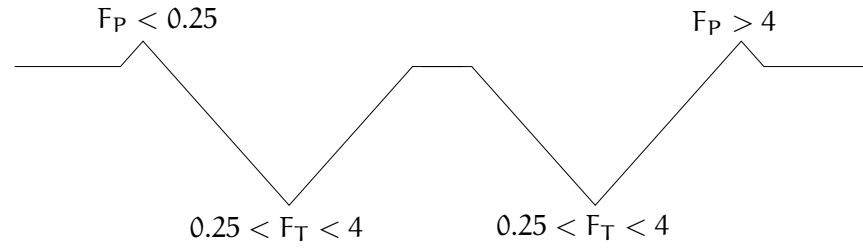
$$F_T = \frac{\text{sum}_{L,t}}{\text{sum}_{R,t}}, \quad (83)$$

where $\text{sum}_{L,t}$ is the sum of the sediment heights left of a trough and $\text{sum}_{R,t}$ is the sum of the sediment heights right of a trough.

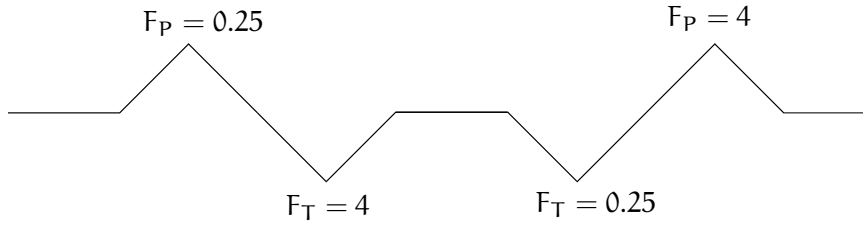
F_P and F_T can only be calculated at locations where the sums between markers are non zero, however in the case that the peak or trough is adjacent to a plateau, the sum on one side will be equal to zero. Thus a different method of judgement must be introduced for these cases. A boolean variable is defined for peaks and troughs, which indicate whether an unclosed peak or trough respectively exists. A peak is deemed unclosed when an ascending peak method is used without a descending peak method, and a trough is deemed unclosed when a descending trough method is used without an ascending trough method. The procedure for determining the appropriate slope method is given in algorithm 2.

There may be some bed configurations that are not well interpreted by the T-factor and P-factor and the respective boolean variables, as the choice made is sometimes dependant on whether the slope is analysed about the peak or the trough. Consider a single narrow peak above a wide bed, where the peak has an indentation. It can be seen that if the first point chosen to be analysed is the first peak before the indentation, the peak will have a T-factor of $F_T < 0.25$ and the slope will be interpreted as the ascending slope of a trough located to the left of the sediment slope. The procedure would then progress to the second peak, where $F_T > 4$. This would cause the rear slope to also be interpreted as a trough-facing, descending slope. As the algorithm currently stands, the ascending slope would be first calculated at the foot of the slope with reference to the current P_{bool} state and would be interpreted as a peak slope, however it is conceivable that some bed configurations would be interpreted in a non-ideal fashion. The indented peak example and some examples of potentially non-ideal interpretations are illustrated in figure 24.

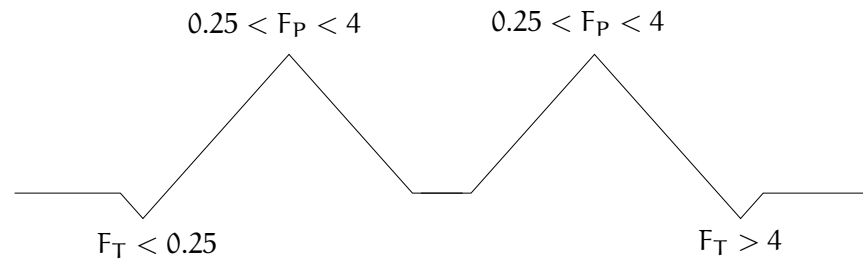
The method for the redistribution of the bed heights of a slope on a prominent peak is as follows. The method first requires the calculation of the gradient between the current peak/trough location marker and the previous marker. If greater than the critical gradient, the ar-



(a)



(b)



(c)

Figure 23: Illustration of P-factor and T-factor. (a) Predominant trough features, (b) equally trough and peak like and (c) predominant peak features

Algorithm 2 : Algorithm for the determination of peak or trough features

Input : Bed heights h , Peak Trough locations PTL, Jump Locations boolean J_b , Plateau Locations boolean P_b , Plateau Lengths PL

Output : Redistributed bed heights h

```

1 Set  $T_{bool}$  and  $P_{bool}$  to false
2 for  $a = 1$  to final value of  $a$ ,  $a := a + 1$  do
3   if  $h[PTL[a]] < h[PTL[a - 1]]$  then
4     See plateau end algorithm 3
5   else if  $h[PTL[a]] < h[PTL[a - 1]]$  then
6     Calculate  $sum_{L,t}$  and  $sum_{R,t}$ 
7     if  $sum_{L,t} = 0$  or  $sum_{R,t} = 0$  then
8       if  $P_{bool} = false$  AND ( $T_{bool} = false$  OR  $T_{bool} = true$ ) then
9         Call Trough Method (descending),  $T_{bool} := true$ 
10      else if  $P_{bool} = true$  then
11        Call Peak Method (descending),  $P_{bool} := false$ 
12      end
13    else
14      Calculate  $F_T$ 
15      if  $F_T < 0.25$  or  $F_T > 4$  then
16        Call Peak Method (descending),  $a := a + 1$ 
17        Call Peak Method (ascending),  $P_{bool} := true$ 
18      else
19        Call Trough Method (descending),  $a := a + 1$ 
20        Call Trough Method (ascending).
21      end
22    end
23  else if  $h[PTL[a]] > h[PTL[a - 1]]$  then
24    Calculate  $sum_{L,p}$  and  $sum_{R,p}$ 
25    if  $sum_{L,p} = 0$  or  $sum_{R,p} = 0$  then
26      if  $T_{bool} = false$  AND ( $P_{bool} = false$  OR  $P_{bool} = true$ ) then
27        Call Peak Method (ascending),  $P_{bool} := true$ 
28      else if  $T_{bool} = true$  then
29        Call Trough Method (ascending),  $T_{bool} := false$ 
30      end
31    else
32      Calculate  $F_p$ 
33      if  $F_p < 0.25$  or  $F_p > 4$  then
34        Call Trough Method (ascending),  $a := a + 1$ 
35        Call Trough Method (descending),  $T_{bool} := true$ 
36      else
37        Call Peak Method (ascending),  $a := a + 1$ 
38        Call Peak Method (descending).
39      end
40    end
41  end
42 end
  
```

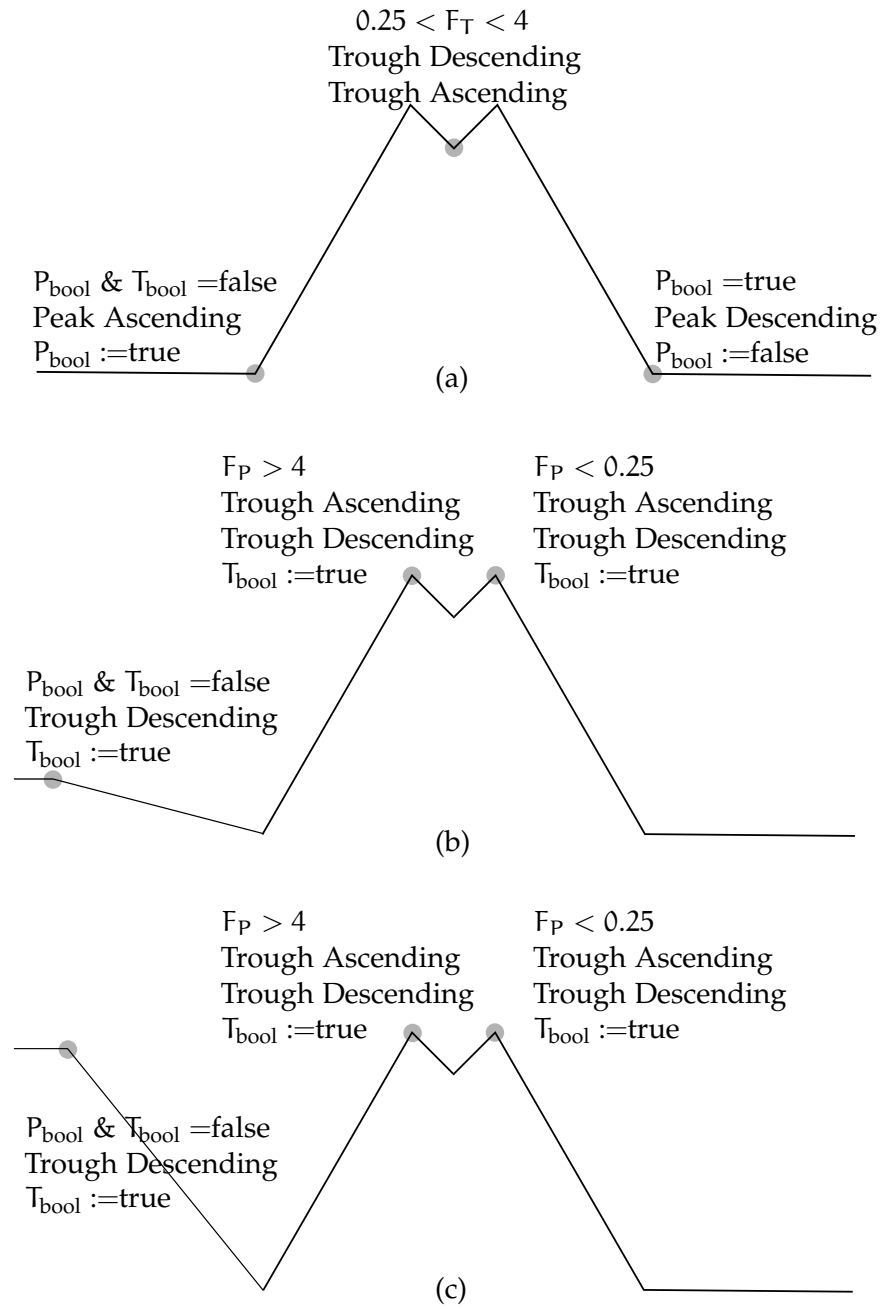


Figure 24: Non ideal interpretation of bed slopes by angle of repose redistribution algorithm. Figure (a) illustrates the interpretation of an indented peak, figure (b) shows a case where the indented peak may be interpreted as a trough and figure (c) shows an alternate form of figure (b), where the trough interpretation is valid. Grey circles mark the points at which slopes were interpreted.

Algorithm 3 : Algorithm for the determination of peak or trough features at the end of a plateau

```

1 if  $P_{\text{bool}} = \text{false}$  AND  $T_{\text{bool}} = \text{false}$  then
2   if  $h[\text{PTL}[a + 1]] > h[\text{PTL}[a - 1]]$  then
3      $a := a + 1$ , Call Peak Method (ascending),  $P_{\text{bool}} := \text{true}$ 
4   else
5      $a := a + 1$ , Call Trough Method (descending),  $T_{\text{bool}} := \text{true}$ 
6   end
7 else if  $P_{\text{bool}} = \text{true}$  then
8   if  $h[\text{PTL}[a + 1]] > h[\text{PTL}[a - 1]]$  then
9      $a := a + 1$ , Call Peak Method (ascending),  $P_{\text{bool}} := \text{true}$ 
10  else
11     $a := a + 1$ , Call Peak Method (descending),  $P_{\text{bool}} := \text{false}$ 
12  end
13 else if  $T_{\text{bool}} = \text{true}$  then
14  if  $h[\text{PTL}[a + 1]] > h[\text{PTL}[a - 1]]$  then
15     $a := a + 1$ , Call Trough Method (ascending),  $P_{\text{bool}} := \text{false}$ 
16  else
17     $a := a + 1$ , Call Trough Method (descending),  $T_{\text{bool}} := \text{true}$ 
18  end
19 end

```

eas filled by the strips of bed are calculated and a triangle equal to this area is constructed using the critical angle of repose. The length of the base and height of the triangle is given by

$$B = \frac{2A}{\sqrt{2A \tan \theta_c}}, \quad (84)$$

$$H = +\sqrt{2A \tan \theta_c}, \quad (85)$$

where B is the base length or the horizontal distance away from the peak, H is the height of the peak above the height of the trough, A is the area of sediment that would lie in the triangle whose gradient is above the critical value and θ_c is the critical angle of repose. The area of sediment A is calculated by summing the difference in height between the height at a point and the trough height, for all points between two Peak Trough Locations. See figure 25 for an illustration of the area sum.

Special consideration must be taken for plateaus located at the peak of a slope. Peaks with plateaus of even length can be evenly split, symmetrically with the peak lying exactly between columns. Plateaus of odd length and simple peaks (i.e. plateaus of zero length) cannot be split as simply. In these cases a symmetrical division of the sediment would require splitting the material of the central width between the smoothing steps for the ascending and descending slopes leading into and out of the peak. Initial forms of the algorithm attempted

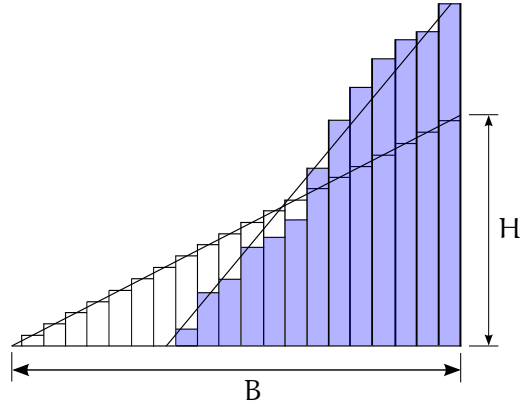


Figure 25: Illustration of the sum of the strip areas, shaded blue, and their newly redistributed heights.

this, however it is difficult to ensure conservation of sediment volume as the division of the centre cell assumes that both slopes will undergo smoothing. If one slope does not require smoothing, half of the volume of the centre width may be lost. It would be possible to account for this if it were known which slopes are to undergo smoothing, however this is not possible as the smoothing of any one slope will impact the bed heights around it, changing neighbouring slopes and gradients.

The solution to this involves the emulation of the even plateau case calculation for odd and zero length plateaus. This requires a non symmetrical analysis, where the ascending slope, for example, may be required to be smoothed up to the penultimate column before the peak, with the peak column being smoothed as part of the smoothing of the descending slope. In an effort to not favour one side more than the other, each time the algorithm is called, it switches from including the centre cell with the right hand slope to including it with the left hand slope. This switching will henceforth be referred to as running the algorithm in an A-sweep (where the centre cell is included on the descending slope from left to right) or B-sweep (where the centre cell is included on the ascending slope from left to right). The sum of the areas can now be defined in equation 86 for ascending slopes, where the current marker, α , is the location marker for a peak, and equation 87 for descending slopes, where the current marker α is the location marker for a trough.

$$A = \sum_{i=\text{PTL}[a-1]}^{i=\text{PTL}[a]+\lfloor \text{PL}[a]/2 \rfloor - Z_1} (h[i] - h[\text{PTL}[a-1]]) \delta x, \quad (86)$$

$$A = \sum_{i=\text{PTL}[a-1]-\lfloor \text{PL}[a-2]/2 \rfloor + Z_2}^{i=\text{PTL}[a]} (h[i] - h[\text{PTL}[a]]) \delta x, \quad (87)$$

where $Z_1 = 1$ for all cases and $Z_2 = 1$ for even length plateaus and $Z_2 = 0$ for odd or zero length plateaus in the A-sweep and $Z_1 = 1$ for even length plateaus and $Z_1 = 0$ for odd or zero length plateaus and $Z_2 = 1$ for all cases in the B-sweep. $\text{PL}[a]$ or $\text{PL}[a-2]$ is the plateau length at the peak for ascending or descending slopes respectively.

Once the area of sediment is calculated and used to calculate the base and height of the new triangle, the height at the peak can be set. The height in the cell nearest the peak (which we now define as lying between cells) is given by equation 88 for ascending slopes and by equation 89 for descending slopes.

$$\begin{aligned} h[\text{PTL}[a] + \lfloor \text{PL}[a]/2 \rfloor + Z_1] &= h[\text{PTL}[a-1]] + H \\ &\quad - 0.5\delta x \tan(\theta_c), \end{aligned} \quad (88)$$

$$\begin{aligned} h[\text{PTL}[a-1] - \lfloor \text{PL}[a-2]/2 \rfloor + Z_2] &= h[\text{PTL}[a]] + H \\ &\quad - 0.5\delta x \tan(\theta_c), \end{aligned} \quad (89)$$

where $Z_1 = -1$ for any case in A-sweep and $Z_1 = -1$ for even plateau lengths and $Z_1 = 0$ for odd or zero length plateaus in B-sweep. $Z_2 = 1$ for even plateau lengths and $Z_2 = 0$ for odd or zero length plateaus in A-sweep and $Z_2 = 1$ for any case in B-sweep.

The heights of the rest of the points that now lie on this new triangle are substituted in. The new heights are easily calculated within the limits of the two location markers in question, see equation 90 for ascending slopes and equation 91 for descending slopes.

$$h[i] = h[i-1] - \delta x \tan(\theta_c), \quad (90)$$

$$h[i] = h[i+1] - \delta x \tan(\theta_c), \quad (91)$$

where h is the discrete height of the point at the horizontal location i .

The nature of the smoothing is likely to cause the new triangle to be wider than the width between the originally calculated peak and trough, see figure 25. The areas in the outer region were not summed in the calculation of the new triangle area and thus by overwriting the height values in the outer region, the volume of sediment would not be conserved. By summing the theoretical additional sediment height

for the calculated triangle with the sediment height that already exists at that point, we can conserve sediment volume, see figure 26. See equation 92 for ascending slopes and equation 92 for descending slopes.

$$h[i] = h[i] + h[PTL[a - 1]] - c\delta x \tan(\theta_c) - O, \quad (92)$$

$$h[i] = h[i] + h[PTL[a]] - c\delta x \tan(\theta_c) - O, \quad (93)$$

where c is a counter which counts the number of columns since passing the lower limit and O is the original height at the lower limit. O must be stored prior to calculation as the height at that location will be overwritten before this calculation is reached.

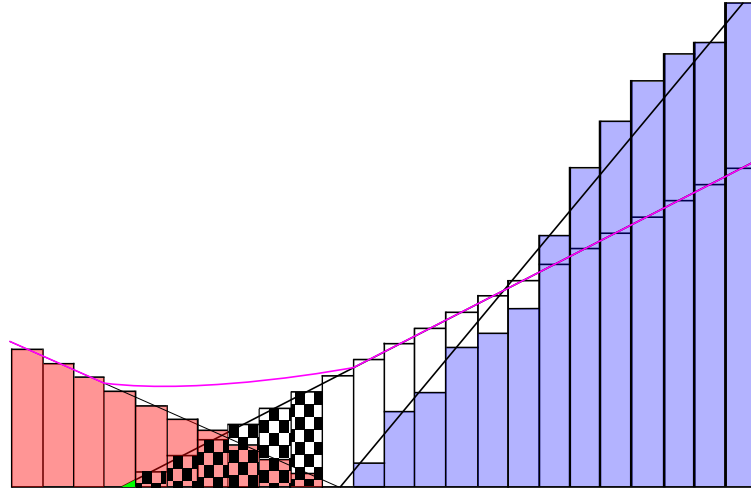


Figure 26: Illustration of the sums required in the region beyond a trough. Areas shaded blue are the areas to be smoothed, areas shaded red are a neighbouring slope, checkered areas are the areas to be summed with the values of the red neighbouring slope and the area in green is the triangular remainder. The magenta line represents the final bed form as a result of the redistribution of the blue slope.

The intersection of the triangle with the base may fall anywhere inside of the final cell. In order to conserve area of sediment, the area of the triangle that exists within the final cell (shaded in green in figure 26) is separately calculated and spread across that final cell by calculating the height equivalent associated with a rectangle of the same area. The equation for the height in this final cell is given by equation 94 for ascending slopes, where the triangular remainder lies beyond the lower limit or trough, equation 95 for ascending slopes, where the triangular remainder lies at or before the lower limit or trough, equation 96 for descending slopes, where the triangular remainder lies beyond the lower limit or trough, and equation 97 for descend-

ing slopes, where the triangular remainder lies at or before the lower limit or trough.

$$\begin{aligned} h[PTL[a] + \lfloor PL[a]/2 \rfloor - B_i + Z_1] &= h[PTL[a] \\ &+ \lfloor PL[a]/2 \rfloor - B_i + Z_1] + (B - B_i \delta x)^2 \tan(\theta_c), \end{aligned} \quad (94)$$

$$h[PTL[a] + \lfloor PL[a]/2 \rfloor - B_i + Z_1] = O + (B - B_i \delta x)^2 \tan(\theta_c), \quad (95)$$

$$\begin{aligned} h[PTL[a-1] - \lfloor PL[a-2]/2 \rfloor + B_i + Z_2] &= h[PTL[a-1] \\ &- \lfloor PL[a-2]/2 \rfloor - B_i + Z_2] + (B - B_i \delta x)^2 \tan(\theta_c), \end{aligned} \quad (96)$$

$$\begin{aligned} h[PTL[a-1] - \lfloor PL[a-2]/2 \rfloor + B_i + Z_2] &= O \\ &+ (B - B_i \delta x)^2 \tan(\theta_c), \end{aligned} \quad (97)$$

where $B_i = \lfloor B/\delta x \rfloor$, $Z_1 = -1$ for any case in A-sweep and $Z_1 = -1$ for even plateau lengths and $Z_1 = 0$ for odd or zero length plateaus in B-sweep. $Z_2 = 1$ for even plateau lengths and $Z_2 = 0$ for odd or zero length plateaus in A-sweep and $Z_2 = 1$ for any case in B-sweep.

There are various advanced conditions that also need to be accounted for. It is not guaranteed that the height of the newly created triangle will be below the height of the peak, particularly in the case where a plateau exists increasing the mass that must be accounted for in the particular smoothing step. In situations where the calculated new height H is above the current peak height, an adaptation must be made to force the height to equal the peak height and increase the base length to account for the additional mass, further reducing the angle of the slope. The physical justification for this condition is to not allow redistribution to cause an increase in potential energy.

Additionally the base width of the new triangle may be longer than the available space, extending outside the control volume or domain. In this situation the area of material that would extend outside the domain is calculated and spread evenly across the entire remaining base. The additional area that falls outside of the domain, and the resulting additional height given to all points between the limit and the domain edge are given by equations 98 and 99.

$$A_E = \left(\left(B - \left(PTL[a] + \left\lfloor \frac{PL[a]}{2} \right\rfloor + Z \right) \delta x \right)^2 \tan(\theta_c) \right) / 2, \quad (98)$$

$$H_E = \frac{A_E}{\left(PTL[a] + \left\lfloor \frac{PL[a]}{2} \right\rfloor + Z \right) \delta x}, \quad (99)$$

where A_E is the area that lies outside the domain, H_E is the additional height added to all points between the domain edge and the upper

limit, $Z = -1$ for all cases in the A-sweep and $Z = -1$ for even length plateaus or $Z = 0$ for odd or zero length plateaus in the B-sweep.

The analogous equations for extra area and height for a descending slope, where $PTL[a]$ represents a trough are

$$A_E = ((B - (Z + n_{cols} - 2 - (PTL[a - 1] + \left\lfloor \frac{PL[a - 2]}{2} \right\rfloor)))^2 \tan(\theta_c)) / 2, \quad (100)$$

$$H_E = \frac{A_E}{(Z + n_{cols} - 2 - (PTL[a - 1] + \left\lfloor \frac{PL[a - 2]}{2} \right\rfloor)) \delta x}, \quad (101)$$

Where $Z = 1$ for odd or zero length plateaus or $Z = 0$ for even length plateaus in the A-sweep. $Z = 0$ for all cases in the B-sweep.

An additional case was also required for peaks located at the right hand boundary, as no plateau length can exist beyond the location. As such, plateau length can not be called and is considered to be zero for this case.

Errors can arise when PTL markers are located in adjacent columns. This is because the width reduction factor Z will, for some cases and sweep types, reduce the width to zero. This will cause a division by zero or a “not a number” (NaN) failure. In cases where the width reduction is not applied, the method does not successfully redistribute the sediment, either by leaving a jump, or by only smoothing a single column. The solution to this is to include a separate method for these jump cases. The method for a peak sums the sediment for successively greater numbers of columns until the height of the new triangle formed by the area is greater than the height above the trough at that location. The preceding column is chosen as the new peak/trough location marker for this slope smoothing step and the method continues as normal, using exactly this point to smooth away from (negating potential plateau length or even/odd direction requirements). The preceding column is chosen, as using the column where the new bed height is greater than the old bed height would cause an increase in bed height and potential energy. The advantage of using this method is that it exactly finds the area needed to create the new slope, however it does not take into account peak symmetry and will create asymmetric smoothed peaks for large area cases.

The method presented above describes the equations for the calculation of new sediment bed heights descending from or ascending to a prominent peak feature. An analogous method can be produced for the slopes around a prominent trough feature. The methods are largely the same, however the area that is to be smoothed is the area of the void above the sediment bed, where the upper bound is the height of the previous or next peak (for descending or ascending slopes respectively). The new height of the triangle H is calculated as above and is applied at the trough centreline by subtracting it

from the appropriate peak height (as opposed to summing it with the trough height, as in equations 88 and 89). All further calculations are completed as above, with the adaptation of the subtractions from the peak height. Note that the case where the new triangle extends beyond the edge of the domain requires the extra height H_E to also be subtracted. See figure 27 for an illustration of the void area calculation and the resulting redistribution smoothed away from the trough centreline.

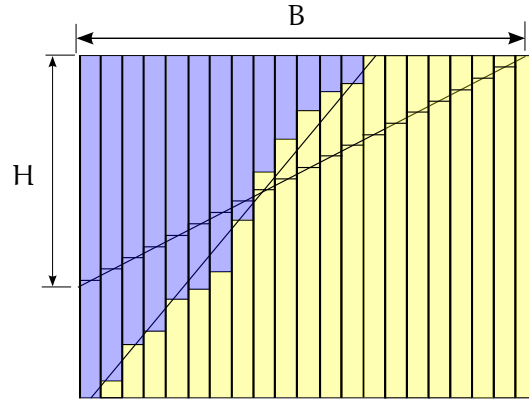


Figure 27: Illustration of the sum of the void strip areas, shaded blue, and their newly redistributed heights. Note that the trough centreline is collocated with the left hand edge of the diagram.

The present algorithm moves across the various slopes, redistributing these slopes in order from left to right. Although an effort has been made to not over-favour the distribution of material at a non-symmetric peak in one direction or the other, the overall sweep direction can still introduce a systematic error. Further study is needed to quantify this error and to study the impact of different heuristic orders, for example ordering by largest slope gradient, highest peak or largest peak to trough height difference.

Care must be taken with cells where the bed height was artificially raised to match the angle of repose condition as they could potentially have had sediment suspended in them, that has now been lost. This has been accounted for by assuming the bed collapsed below this material effectively scooping up all of the material that were in the cells that now form the bed into the cell immediately above the new top bed cell. Even if no material must be moved upwards, the change in cell area at the bed surface will cause a change in concentration, if conservation of mass is to be maintained.

The present method considers redistribution to be instantaneous and does not model the sediment particle's gain in kinetic energy from their loss of potential energy as they roll down the slope. Constant maintenance of the angle of repose will cause any redistribution to only produce small motions and as such the impact of inertia on the sediment particle velocities and final bed heights is assumed to be small.

2.3.7 Continuity

The main condition that must be maintained in the sediment transport calculation is conservation of sediment mass. It can be noted that while the mass and volume of sediment should be conserved, the concentration of suspended sediment C is not a conserved quantity. This is both because suspended sediment concentration can be lost in the process of deposition, but also because the change in bed height can affect the volume of fluid available to the fluid bulk, and thus the concentration of the sediment in that fluid bulk.

Conservation of mass can be monitored by summing the product of the cell concentrations and cell areas for all fluid cells, with the mass equivalent of each width's bed height.

The main condition that must be maintained in the free surface calculation is conservation of fluid volume. When sediment is not considered, this condition can be monitored through a comparison of the sum of the volume fractions for all cells.

The inclusion of sediment means that the sum of the volume fractions no longer monitors the total volume of fluid as some of the fluid is located in the bed where it is effectively trapped from simulation. When all sediment to be included is present at the beginning of the simulation, either in the form of bed height or suspended sediment concentration, conservation of fluid volume can be monitored by summing the volume fraction minus the equivalent sediment volume for each cell.

In the case of a simulation where new mass is being introduced, an increase in volume height must be added to account for the displacement caused by the additional sediment volume. Conservation of volume can then be monitored as in the case of a simulation with constant sediment mass. Conservation of mass is monitored in this case by comparing the new total sediment mass with the sum of the old sediment mass and the mass of the newly introduced material.

2.4 STABILITY CONDITIONS

The time step size (δt) must be chosen to maintain the stability of the numerical scheme. One of two approaches can be taken, one can

either choose a conservative value that will apply for all times or the time step size can be chosen adaptively for each time step. The advantage of an adaptive time stepping scheme is that it will allow for a more efficient choice of δt and thus will reduce computation time. Tome and McKee [1994] discuss the stability conditions that have been applied in the past. The first condition that must be applied is the condition that the fluid cannot pass through two cell boundaries in one time-step, as the differences assume fluxes between adjacent cells only, i.e.

$$\delta t < \frac{\delta x}{|u|_{\max}},$$

$$\delta t < \frac{\delta y}{|v|_{\max}}.$$

Tome and McKee [1994] noted that $|u|_{\max}$ and $|v|_{\max}$ are somewhat ambiguous and could refer to u^n or u^{n+1} . It is significantly easier and computationally less intensive to base the time step size on u^n , however it is not guaranteed that the condition is observed in that time step. For simplicity it was chosen to use the u^n and v^n values, especially since a safety factor will be imposed on the final time step size.

The second condition requires that in cases of non zero kinematic viscosity, that momentum must not diffuse across more than one cell width or height in a time step, i.e.

$$\frac{2\delta t}{\text{Re}} < \frac{\delta x^2 \delta y^2}{\delta x^2 + \delta y^2} = \left(\frac{1}{\delta x^2} + \frac{1}{\delta y^2} \right)^{-1}.$$

In the case of running a simulation without sediment, the time step size may be calculated using only these three conditions, i.e.

$$\delta t = \tau_{\delta t} \min \left(\frac{\text{Re}}{2} \left(\frac{1}{\delta x^2} + \frac{1}{\delta y^2} \right)^{-1}, \frac{\delta x}{|u|_{\max}}, \frac{\delta y}{|v|_{\max}} \right), \quad (102)$$

where $\tau_{\delta t}$ is the time safety factor which ensures that the chosen time step size is smaller than the condition from which it was derived.

However in the case of the inclusion of sediment two more stability conditions must be met. Firstly sediment must not be able to settle through more than one cell height within a time step, i.e.

$$\delta t < \frac{\delta y}{(-v'_s/U')}$$

Secondly sediment must not be able to diffuse through more than one cell width or height within a time step.

$$\delta t < \frac{\delta x^2}{2K_{x,\max}}$$

$$\delta t < \frac{\delta y^2}{2K_{y,\max}}$$

Thus the time step size for simulations including sediment is defined as

$$\delta t = \tau_{\delta t} \min \left(\frac{\text{Re}}{2} \left(\frac{1}{\delta x^2} + \frac{1}{\delta y^2} \right)^{-1}, \frac{\delta x}{|u|_{\max}}, \frac{\delta y}{|v|_{\max}}, \frac{\delta y}{(-v'_s/U')}, \frac{\delta x^2}{2K_{x,\max}}, \frac{\delta y^2}{2K_{y,\max}} \right).$$

In a worst case scenario, it is possible that the conditions given above are met whilst the simulation becomes unstable, causing the mass of sediment to not be conserved. This is because the conditions ensure the time step is small enough to avoid a problem for each of the conditions in isolation, however it is possible, for example, that the sum of diffusion and advection aligned in the same direction could cause a sediment flux greater than a cell width or height within the time step. In an attempt to maintain the stability of the sediment transport model, a combined stability condition was constructed to represent the worst case scenario in the horizontal direction (advection and diffusion aligning) and in the vertical direction (settling velocity, advection and diffusion aligning). Thus the horizontal and vertical joint stability conditions are

$$\delta t < \frac{1}{\frac{2K_{x,\max}}{\delta x^2} + \frac{|u|_{\max}}{\delta x}},$$

$$\delta t < \frac{1}{\frac{2K_{y,\max}}{\delta y^2} + \frac{|v|_{\max}}{\delta x} + \frac{(-v'_s/U')}{\delta y}}.$$

The combined stability condition ensures that stability is maintained, however this is paid for by a reduction in computational speed, caused by the use of an overly conservative time step size. In an attempt to mitigate this, an alternative algorithm for the dynamic selection of time step size was constructed. The algorithm calculates the time step size associated with each of the various stability conditions, takes the smallest of the velocity and Reynolds numbers conditions (i.e. the standard condition taken for a simple, no sediment run of the [NS](#) solver) and uses this as a starting point. If this value is insufficiently small it returns to the dynamic time step calculator, which moves through a sorted list of the remaining stability conditions, from largest to smallest, until the program succeeds in the time step. At the point of success the identifier of the stability condition that succeeded is stored and this condition is used as the starting point for the calculation in the next time step.

An alternative method is to add to the list of conditions values half way between each condition to try to hone in on value of δt that would be both stable and efficient. The final variation of this algorithm is the allowance for the program to reset its starting point time step size whenever a given amount of time or time-steps passes. This

allows for the situation where the program had to resort to the worst case scenario condition however the situation then improved and became better behaved and a larger time step size would be successful.

Algorithm 4 : Peak, trough, plateau and jump location algorithm

Input : h
Output : PTL, J_b , P_b , PL

```

1  if  $h[i] = h[i - 1]$  then
2      if previous gradient is NOT zero then
3          Plateau Start location:  $i-1$ 
4          if  $i \text{ NOT} = 2$  AND  $i \text{ NOT} = \text{PTL}[\text{PTLsize}-1] + 1$  then
5              PTL:  $i-1$ , PL: 0,  $P_b$ : false, if PreviousJump=false then
6                   $J_b$ : true, PreviousJumpb: false.
7              else
8                   $J_b$ : false, PreviousJumpb: false.
9              end
10         end
11     end
12 else if (Previous gradient is positive OR zero) AND
     $h[i] - h[i - 1] > dx$  then
13     PreviousJumpb: true, if Previous gradient is zero then
14         PTL:  $i$ , PL:  $(i - 1) - (\text{plateaustart} - 1)$ ,  $P_b$ :true,  $J_b$ :true
15     else
16         PTL:  $i - 1$ , PL: 0,  $P_b$ :false,  $J_b$ :true
17     end
18 else if (Previous gradient is negative OR zero) AND
     $h[i - 1] - h[i] > dx$  then
19     PreviousJumpb: true, if Previous gradient is zero then
20         if  $i \text{ NOT} = 2$  then
21             PTL:  $i - 1$ , PL:  $(i - 1) - (\text{plateaustart} - 1)$ ,  $P_b$ :true,
                 $J_b$ :false
22         else
23             PTL:  $i - 1$ , PL: 0,  $P_b$ :true,  $J_b$ :false
24         end
25     end
26 else if  $h[i] \text{ NOT} = h[i - 1]$  AND previous gradient is zero then
27     PTL:  $i - 1$ , PL:  $(i - 1) - (\text{plateaustart} - 1)$ ,  $P_b$ :true,  $J_b$ :false
28 else if (Previous gradient is positive) AND  $h[i] < h[i - 1]$  then
29     PTL:  $i - 1$ , PL: 0,  $P_b$ :false, if PreviousJumpb=true then
30          $J_b$ : true, PreviousJumpb: false
31     else
32          $J_b$ :false, PreviousJumpb: false
33     end
34     if  $h[i - 1] - h[i] > dx$  then
35         PreviousJumpb: true
36     end
37 else if (Previous gradient is negative) AND  $h[i] > h[i - 1]$  then
38     PTL:  $i - 1$ , PL: 0,  $P_b$ :false, if PreviousJumpb=true then
39          $J_b$ : true, PreviousJumpb: false
40     else
41          $J_b$ :false, PreviousJumpb: false
42     end
43     if  $h[i] - h[i - 1] > dx$  then
44         PreviousJumpb: true
45     end
46 end

```

NUMERICAL VALIDATION

3.1 NAVIER-STOKES SOLVER EFFICIENCY INVESTIGATION

The iterative methods used to solve the [PPE](#) were tested for their efficiency using a lid-driven cavity simulation. A lid-driven cavity is a filled square domain with no-slip conditions on the lower, left hand and right hand boundaries and a prescribed tangential non-dimensional velocity equal to 1 at the upper boundary. It is a standard test problem used to analyse the performance of Navier-Stokes solvers. The results given by [Ghia et al. \[1982\]](#) are used for comparison. Figures [28](#) and [29](#) illustrate the solution of the lid-driven cavity problem for a Reynolds number equal to 100 on a 128×128 grid.

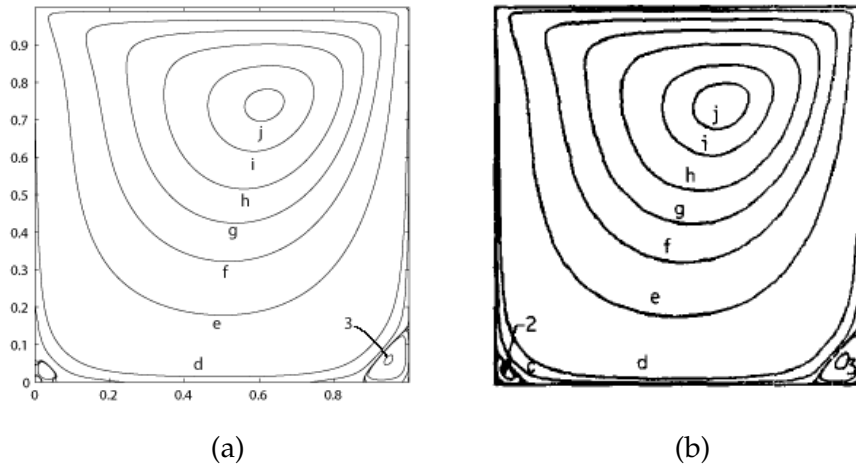


Figure 28: Comparison of spun up streamline plots for a lid-driven cavity problem ($Re = 100$) for (a) the present model and (b) the results presented by [Ghia et al. \[1982\]](#). Contour values: $a = -1 \times 10^{-10}$, $b = -1 \times 10^{-7}$, $c = -1 \times 10^{-5}$, $d = -1 \times 10^{-4}$, $e = -0.01$, $f = -0.03$, $g = -0.05$, $h = -0.07$, $i = -0.09$, $j = -0.1$.

The simulation was run from a zero field initial condition for the velocity and pressure for a [ND](#) time $t = 10$. Tests were conducted on various grid sizes, for flows with a range of velocity scales and thus Reynolds numbers. The methods were judged by studying the number of iterations to convergence and the computational time, or clock ticks, for each time step and the cumulative number of clock ticks up to a given point which is analogous to the total computational run time to the given point. The maximum allowable number of iterations is set to the grid resolution squared.

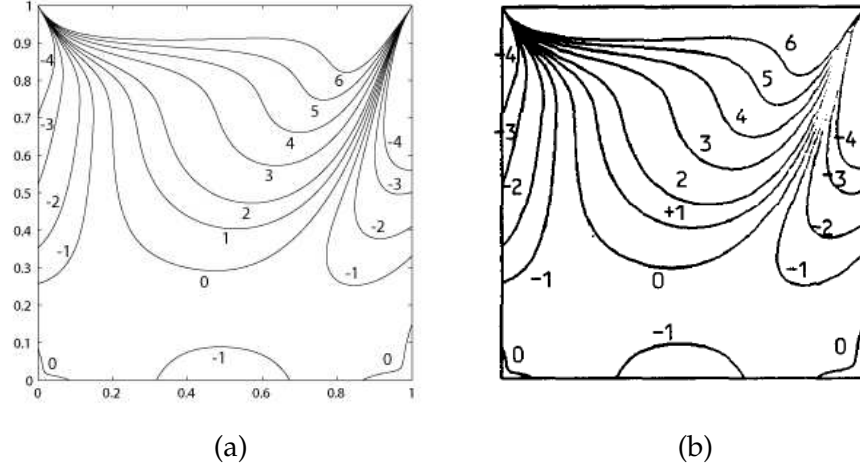


Figure 29: Comparison of spun up vorticity plots for a lid-driven cavity problem ($Re = 100$) for (a) the present model and (b) the results presented by Ghia et al. [1982]. Contour values: $-1 = -0.5$, $0 = 0$, $1 = 0.5$, $2 = 1$, $3 = 2$, $4 = 3$, $5 = 4$, $6 = 5$.

Primary tests on a very coarse grid showed that both forms of the Gauss-Seidel method performed less well, with successive over relaxation, conjugate gradients and the multigrid method performing significantly better, see figure 30. The MG method required the fewest number of iterations, however SOR and CG both performed better in time as MG has multiple iterations or smoothing steps for each iteration of the method. It must be noted that the very coarse resolution grids are unable to fully encapsulate the flow at higher Reynolds numbers causing failure to properly resolve.

Increasing the grid resolution causes the two forms of Gauss-Seidel to perform even less well, widening the gap between the performance of the GS method and the performance of the other three methods, in terms of iterations, time and variability. As grid resolution increased, SOR performed significantly worse both in the number of iterations/speed as well as in increasing variability between steps. Increasing the grid resolution has little impact on the performance of both the MG and CG methods in terms of number of iterations, however the increase in grid resolution does increase the time taken per step. See figures 31 and 32.

Figures 33, 34, 35, 36 and 37 show the impact of grid size on each of the methods separately. The GS, RBGS and SOR methods all reach the maximum allowable iteration limit, and thus the comparisons are no longer comparing iterations to reach convergence, as convergence was not reached in the fine grid scenarios. It is possible, however, to compare the time taken to reach the limit for different grid resolutions that both reached the limit. In order to accomplish this, the maximum number of iterations limit must be set to a single arbitrary value (1000 steps) used by the methods on different grid sizes. The GS, RBGS and

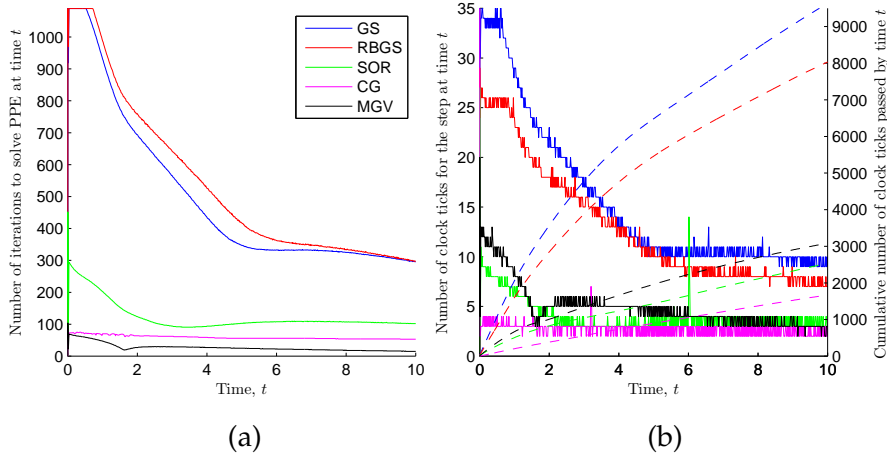


Figure 30: Impact of the solution method on the efficiency of the Navier-Stokes solver on a 33×33 grid, where $Re = 1000$. Figure (a) shows the number of iterations required to reach convergence for a given time step. Figure (b) shows the time taken to complete the step in clock ticks (solid lines) and the cumulative number of clock ticks up to that point (dashed lines). All following figures within section 3.1 are presented in this two panel form.

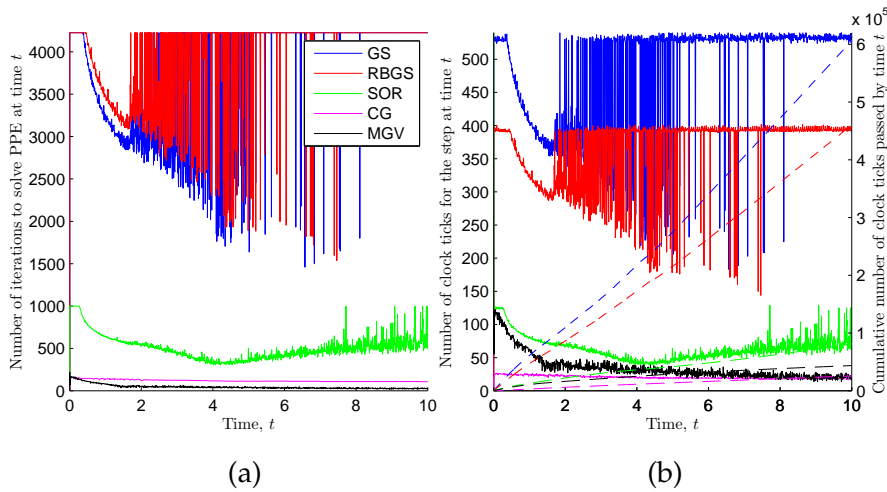


Figure 31: Method comparison on a 65×65 grid, where $Re = 1000$.

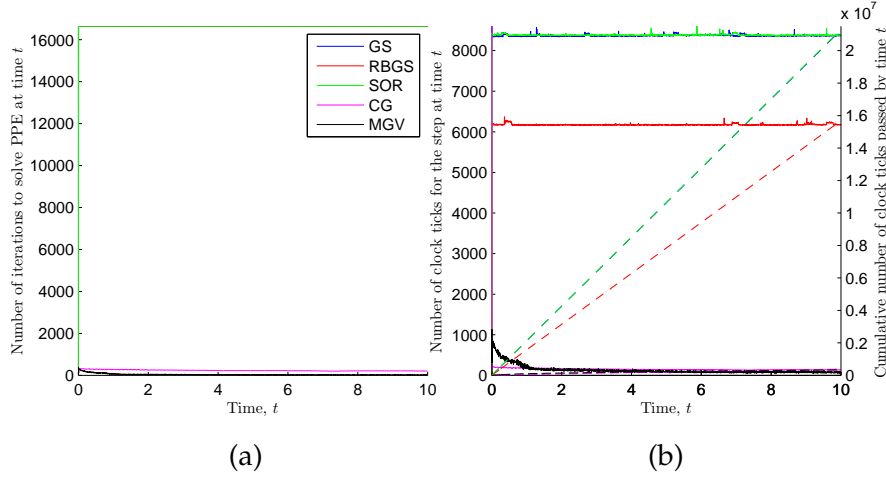


Figure 32: Method comparison on a 129×129 grid, where $Re = 1000$.

SOR methods show a quadrupling of time taken to complete a step for a doubling of grid resolution.

The GS, RBGS, SOR and MG methods all show a trend towards a reducing iteration requirement as the simulation progresses. The CG method, however, shows a relatively constant iteration requirement for the entire simulation. The number of iterations required doubles when the grid resolution doubles. However, doubling of grid resolution causes a tenfold increase in clock ticks per step.

Figure 37 shows that grid size has very little overall impact on the number of iterations required to reach convergence. It can be seen, however, that the grid size does impact the number of iterations required at the initiation of the simulation, with larger grids requiring greater number of iterations at initiation. This is matched by significant increase in time taken per step at the initiation, however after an initial period the difference in both number of iterations and time taken per step reduces significantly, with a doubling of grid resolution causing almost no increase in number of iterations but a quadrupling of time taken per step.

The impact of the Reynolds number on the simulation was investigated. This was completed by varying the characteristic velocity of the flow. Note that these tests were completed on a coarse grid to ensure all methods converged in reasonable time. All methods showed a general trend that increasing the characteristic velocity, and thus the Reynolds number, reduced the number of iterations per step, the time taken per step and the cumulative time taken overall, see figure 38 for the results produced by the MG method. The simulations conducted at a Reynolds number of 100 began the simulation with a higher iteration requirement, which follows the pattern, however went on to finish the simulation with the lowest iteration per step requirement. This is due to the impact of the Reynolds number on the time taken for the simulation to “spin up”. The two Gauss-Seidel methods showed

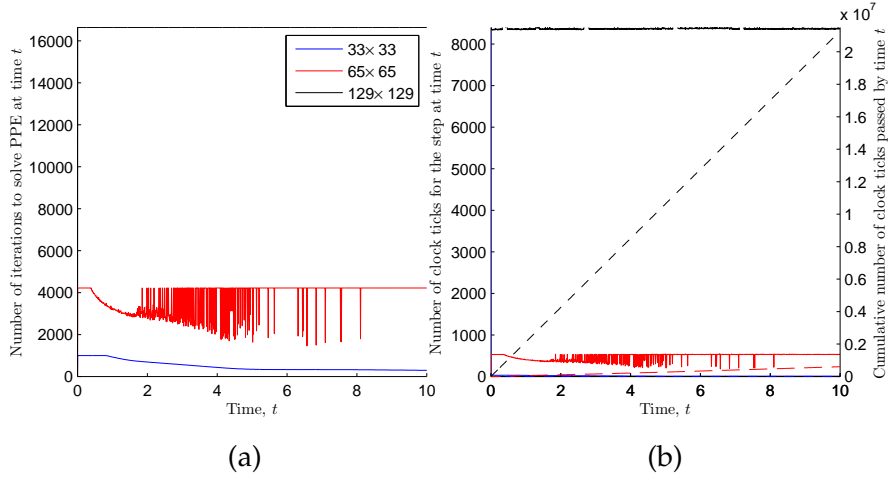


Figure 33: Impact of the grid size on efficiency for the GS method, where $Re=1000$.

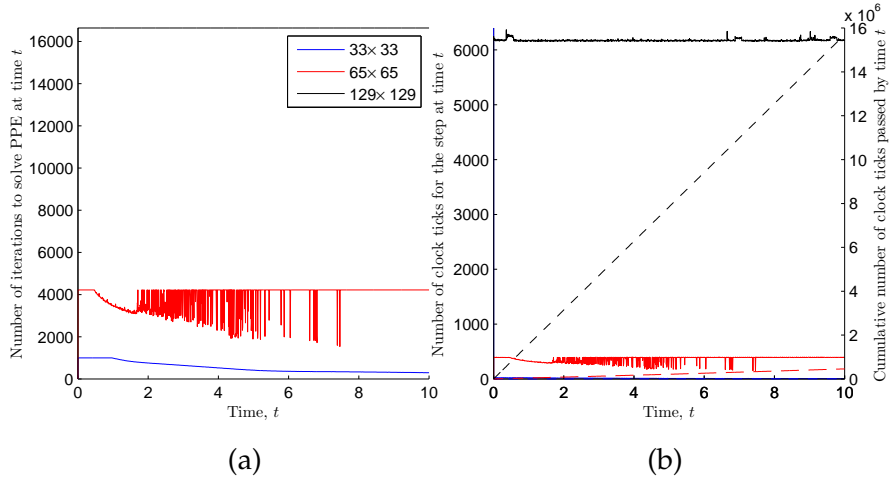


Figure 34: Impact of grid size on efficiency for the RBGS method, where $Re=1000$.

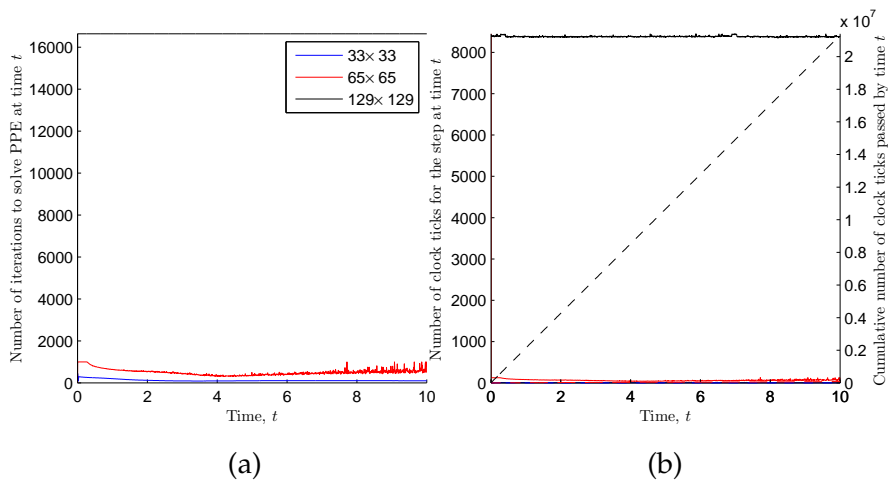


Figure 35: Impact of grid size on efficiency for the SOR method, where $Re=1000$.

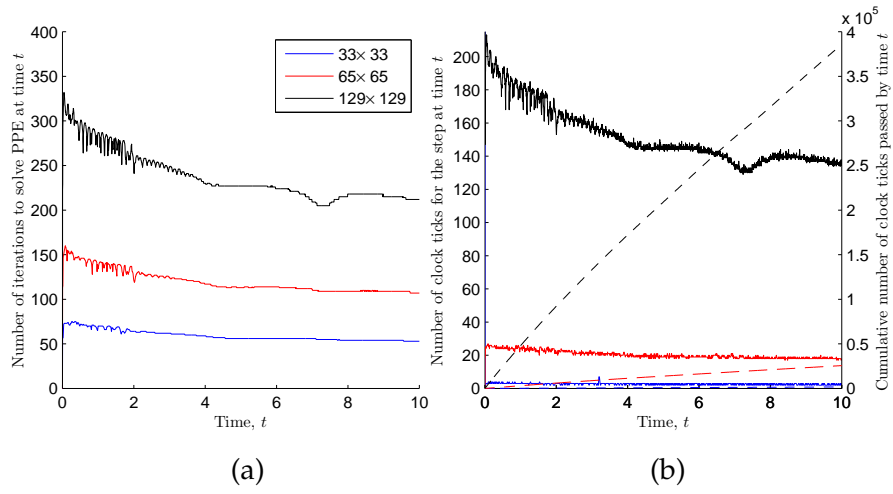


Figure 36: Impact of grid size on efficiency for the CG method, where $Re=1000$.

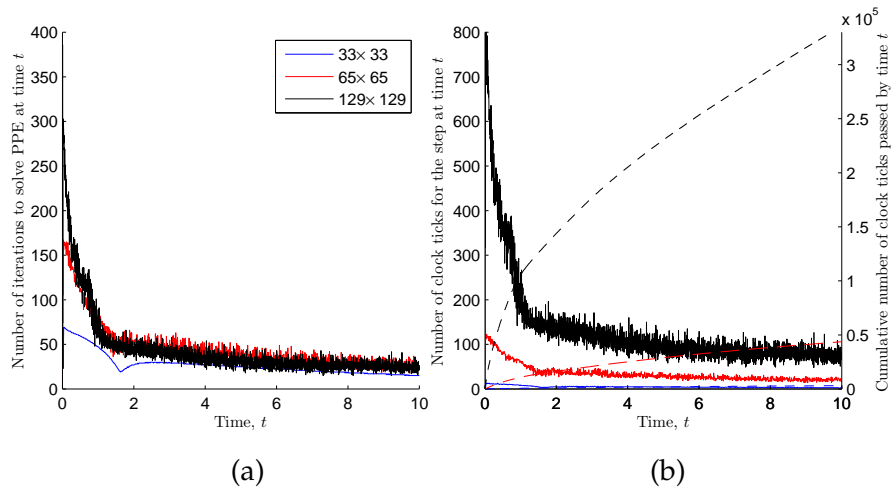


Figure 37: Impact of grid size on efficiency for the MG method, where $Re=1000$.

a similar trend, however they also exhibited increased variability for the $Re = 100$ test.

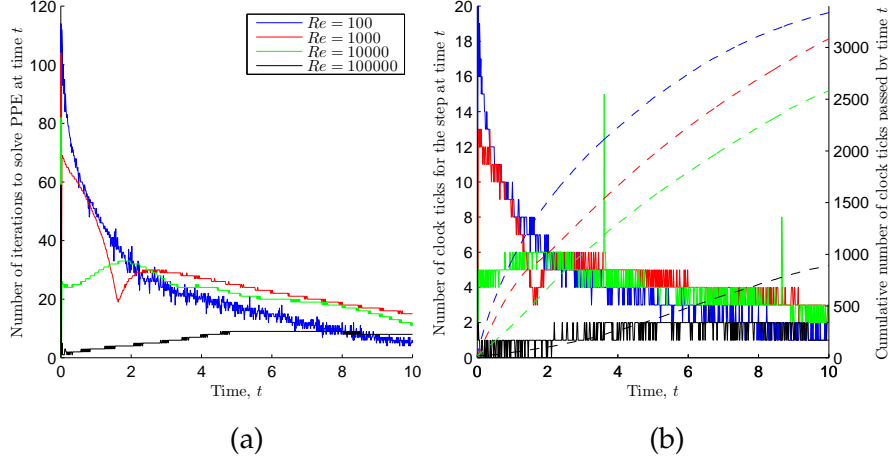


Figure 38: Impact of Reynolds number on efficiency (MG method on a 33×33 grid).

A more in depth look was taken to compare the efficiency of the MG method for variations in grid size and Reynolds number, see figure 39. The first thing to note is the number of iterations all lie within a very narrow region. The figure indicates trends of increasing iteration requirement for increased grid size (for constant Reynolds number) and increasing iteration requirement for a decrease in Reynolds number (for constant grid size). The grouping of the results, however, indicates that the grid size and Reynolds number do not cause significant change to the overall iteration requirement. Figure 39 also shows that the variation of the number of iterations required to converge, from step to step, decreases with increasing Reynolds number. These trends are approximately mirrored by the result for the time taken per step. In general, simulations conducted with a lower Reynolds number had a larger overall simulation time.

The MG method requires smoothing iterations at each grid level. Increasing the number of smoothing iterations (γ_{MGV}) will reduce the number of times the method requires iteration as a whole, however, increasing the number of smoothing iterations increases the computational time per method iteration. The number of smoothing iterations taken at each grid level was investigated so as to determine the most efficient form of the method. Figure 40 shows that increasing the number of smoothing iterations reduces the overall number of iterations required to reach convergence for that time step. The figure showing the computational time or clock ticks per step shows that the larger numbers of smoothing iterations cause an increase in computational time. Note that the lowest number of smoothing steps did not result in the lowest computational time as they require an increase in overall method iterations to reach convergence, increasing the computational

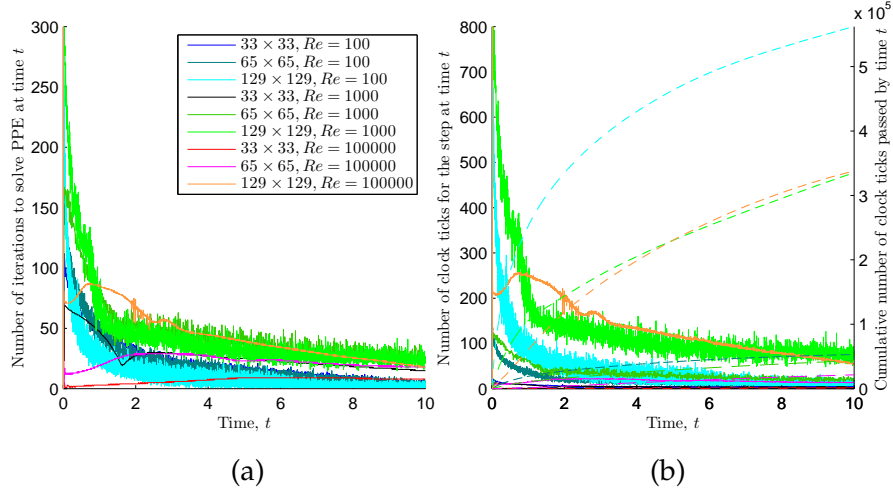


Figure 39: Impact of grid size and Reynolds number on efficiency for the MG method on three grid sizes (33×33 , 65×65 and 129×129), for three Reynolds numbers ($Re = 100$, $Re = 1000$ and $Re = 100000$).

time. A value of γ_{MGV} equal to 3 can be seen to produce the quickest computational speeds, however, values of γ_{MGV} close to this value (2 or 4) produce very similar results.

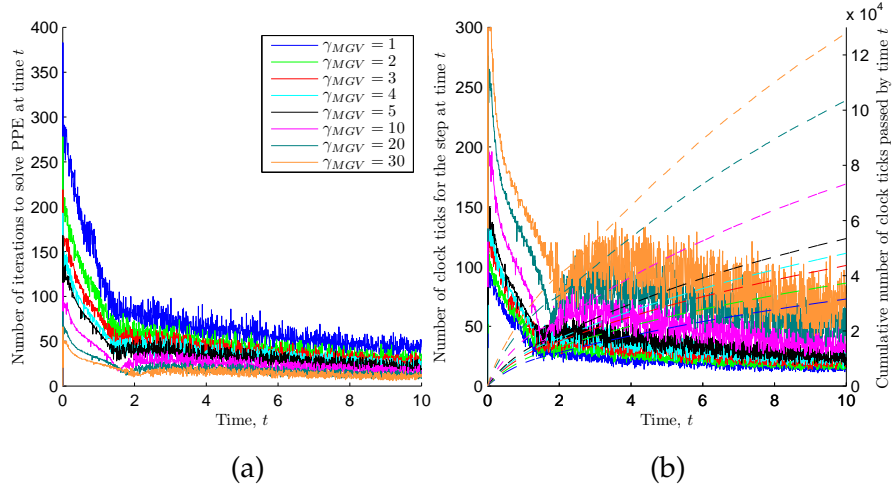


Figure 40: Impact of number of MG smoothing steps, γ_{MGV} , on efficiency (65×65 grid, $Re = 1000$).

The ω factor for the SOR method was investigated. It can be seen in figure 41 that the best performance was achieved by using an ω factor of 1.8, however, the difference in performance for factors of 1.8 and 1.7 is very slight.

The time safety factor was investigated for all methods. All methods showed the trend that reducing the time safety factor causes a reduction in number of iterations and time per step, however, by reducing the time safety factor, the simulation must complete a larger number of steps, reducing the performance of the simulation as a whole. This can be seen for the Gauss-Seidel method in figure 42.

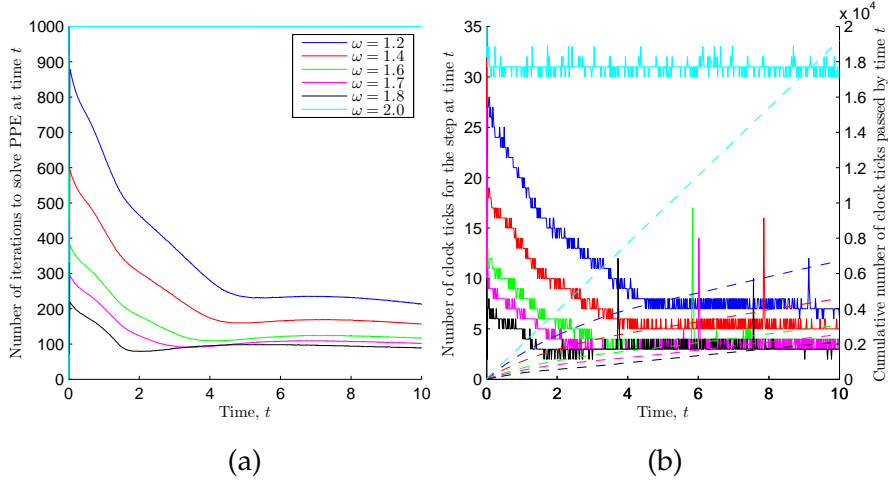


Figure 41: Impact of SOR factor ω on efficiency (33×33 grid, $Re = 1000$).

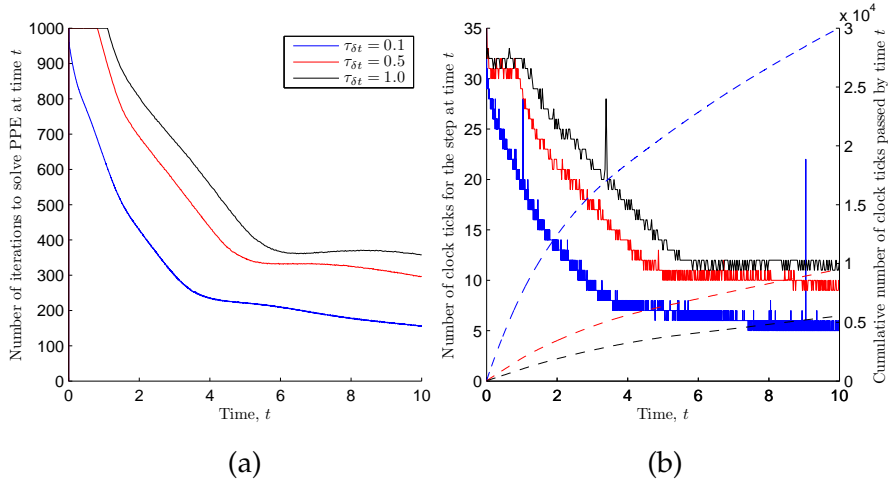


Figure 42: Impact of time safety factor $\tau_{\delta t}$ on efficiency (CS method, 33×33 grid, $Re = 1000$).

The γ safety factor was investigated for all methods. The γ safety factor was found to not significantly impact the performance of the simulation, however the higher γ safety factors, that pushed γ closer to 1 caused a slight improvement in performance for lid-driven cavity simulations. This can be seen for the Gauss-Seidel method in figure 43.

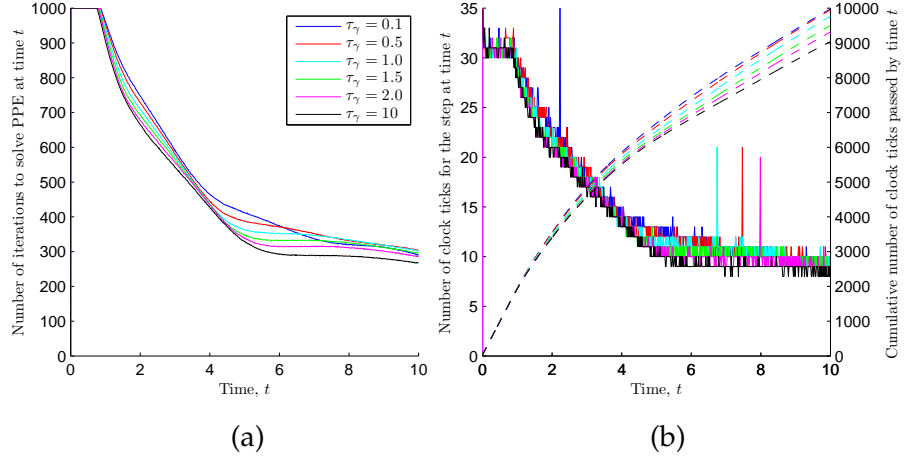


Figure 43: Impact of spatial discretisation safety factor τ_γ on efficiency (GS method, 33×33 grid, $Re = 1000$).

3.2 FREE SURFACE TRACKING TESTS

The current model uses an adapted form of the method presented by Youngs [1982] to track the motion of the fluid and its free surface by maintaining a field of volume fractions. This method was validated against the simpler SLIC and FLAP methods described in appendix A. The methods were tested by imposing a variety of simple velocity fields onto a number of fluid shapes. Figure 44 demonstrates the result of a Youngs' method reconstruction of the free surface of a slotted disk, or Zalesak disk [Zalesak, 1979], and a crescent. It can be seen that the fidelity of the reconstruction is generally very good. The fidelity, however, deteriorates at sharp corners, a trait shared by the reconstructions produced via the SLIC and FLAP methods.

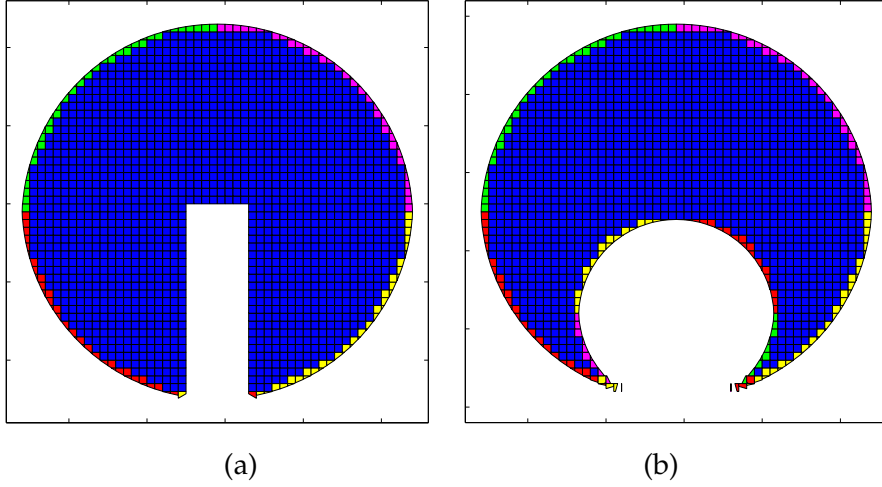


Figure 44: Example of Youngs' reconstruction of the initial conditions for (a) a slotted disk and (b) a crescent, where A cells are shown in green, B cells are shown in magenta, C cells are shown in yellow, D cells are shown in red and filled cells are shown in blue (see 13).

An error metric was computed for each method, determined by summing the difference between the computed solution and the exact solution for each cell over the entire grid and dividing this by the sum of the actual/initial volume fraction field. The general equation for the error is

$$E = \frac{\sum_{i,j=1}^{i_{\max},j_{\max}} \|X_{i,j}^n - X_{i,j}^e\|}{\sum X_{i,j}^{0/e}}, \quad (103)$$

where X^n is the computed quantity after n time steps, X^e is the exact solution, $X^{0/e}$ is the exact solution or the initial condition and $X = c$ for this set of tests. All tests were computed on a 200×200 grid (202×202 including ghost rows and columns), over a domain that went from zero to π in both directions.

3.2.1 Translation

The first test used unidirectional velocity fields to test the method's ability to track translational movements, both in the orthogonal and diagonal directions. The shapes used to test translational motion were a hollow circle (external diameter $\pi/4$, internal diameter $\pi/8$), a hollow rotated square (external width $\pi/4$, internal width $\pi/8$, rotation 0.45 radians) and a crescent (dimensions identical to the hollow circle where the removed circle was located such that its edge contacted the edge of the larger circle at its lowest point). Eight unidirectional velocity fields were applied to create horizontal translation, $(0.5, 0)$ and $(-0.5, 0)$, vertical translation, $(0, 0.5)$ and $(0, -0.5)$, and diagonal translation $(0.5, 0.5)$, $(-0.5, -0.5)$, $(-0.5, 0.5)$ and $(0.5, 0.5)$. The result of an example orthogonal and diagonal translation can be seen in figures 45 and 46 respectively. Note that the images show the initial conditions of the testing object placed in final position of the translation for comparison. The error values for translation can be found in tables 3 and 4.

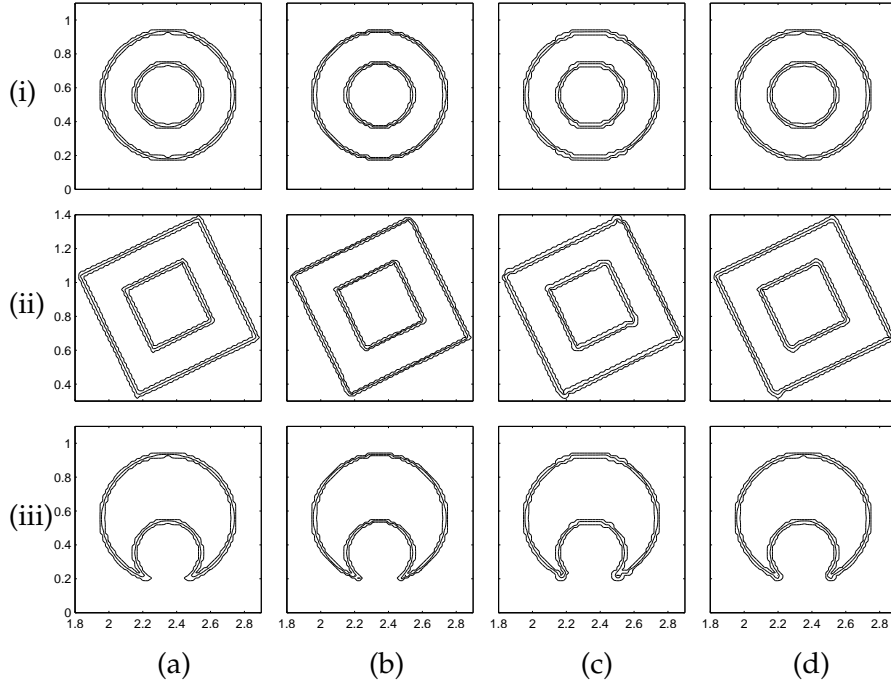


Figure 45: Result of 400 steps of translation in the uniform, unidirectional velocity field $(0.5, 0)$ for (i) a hollow circular test object, (ii) a hollow rotated square test object and (iii) a crescent test object. (a) Initial conditions, (b) SLIC (c) FLAP and (d) YVOF.

The results of the orthogonal translation test show that all three methods maintain the fidelity of the original shape to a high degree, with the Youngs' method consistently producing the nearest result. The diagonal translation saw a distinct reduction in fidelity for all

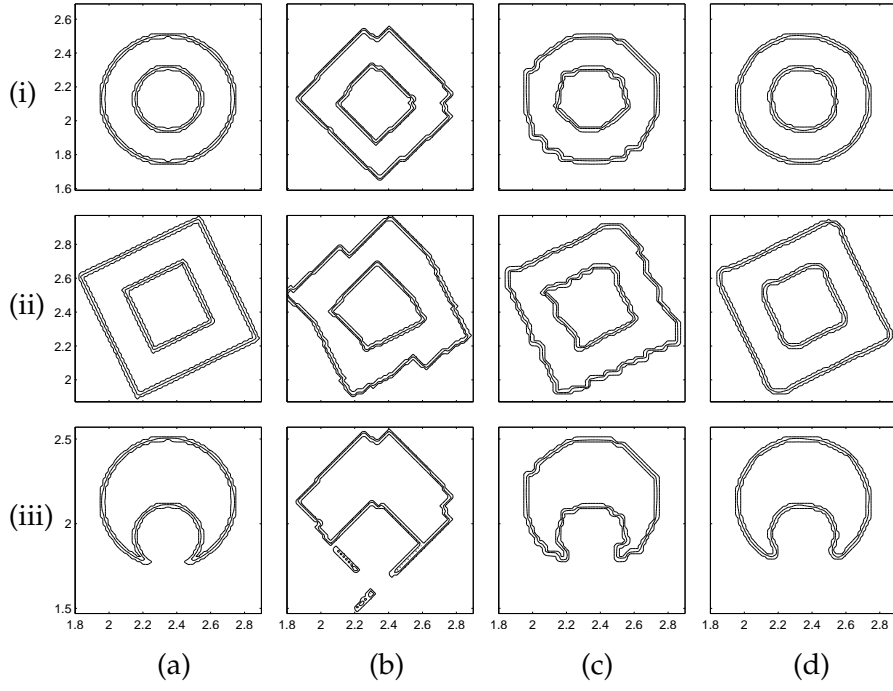


Figure 46: Result of 400 steps of translation in the uniform, unidirectional velocity field $(0.5, 0.5)$ for (i) a hollow circular test object, (ii) a hollow rotated square test object and (iii) a crescent test object. (a) Initial conditions, (b) **SLIC** (c) **FLAP** and (d) **YVOF**.

| Shape | (u, v) | SLIC | FLAP | YVOF |
|---------------|-------------|-----------------------|-----------------------|----------------------|
| Hollow Circle | $(0.5, 0)$ | 1.18×10^{-2} | 1.83×10^{-2} | 1.6×10^{-3} |
| | $(-0.5, 0)$ | 1.18×10^{-2} | 1.83×10^{-2} | 1.6×10^{-3} |
| | $(0, 0.5)$ | 1.16×10^{-2} | 1.41×10^{-2} | 1.7×10^{-3} |
| | $(0, -0.5)$ | 1.6×10^{-2} | 1.41×10^{-2} | 1.6×10^{-3} |
| Hollow Square | $(0.5, 0)$ | 1.31×10^{-2} | 2.20×10^{-2} | 4.4×10^{-3} |
| | $(-0.5, 0)$ | 1.31×10^{-2} | 2.20×10^{-2} | 4.7×10^{-3} |
| | $(0, 0.5)$ | 1.15×10^{-2} | 1.22×10^{-2} | 4.1×10^{-3} |
| | $(0, -0.5)$ | 1.13×10^{-2} | 1.20×10^{-2} | 4.4×10^{-3} |
| Crescent | $(0.5, 0)$ | 8.1×10^{-2} | 1.63×10^{-2} | 4.3×10^{-3} |
| | $(-0.5, 0)$ | 8.1×10^{-2} | 1.63×10^{-2} | 4.3×10^{-3} |
| | $(0, 0.5)$ | 1.27×10^{-1} | 1.62×10^{-2} | 8.4×10^{-3} |
| | $(0, -0.5)$ | 1.27×10^{-1} | 1.90×10^{-2} | 5.6×10^{-3} |

Table 3: Orthogonal Translation Errors

| Shape | (u, v) | SLIC | FLAP | YVOF |
|---------------|-----------|-----------------------|-----------------------|-----------------------|
| Hollow Circle | (0.5, 0) | 2.46×10^{-1} | 7.83×10^{-2} | 3.09×10^{-2} |
| | (-0.5, 0) | 2.46×10^{-1} | 7.83×10^{-2} | 3.09×10^{-2} |
| | (0, 0.5) | 2.46×10^{-1} | 7.83×10^{-2} | 3.09×10^{-2} |
| | (0, -0.5) | 2.46×10^{-1} | 7.83×10^{-2} | 3.09×10^{-2} |
| Hollow Square | (0.5, 0) | 1.80×10^{-1} | 7.75×10^{-2} | 2.70×10^{-2} |
| | (-0.5, 0) | 1.75×10^{-1} | 8.53×10^{-2} | 2.49×10^{-2} |
| | (0, 0.5) | 1.96×10^{-1} | 7.15×10^{-2} | 2.51×10^{-2} |
| | (0, -0.5) | 2.01×10^{-1} | 6.62×10^{-2} | 2.56×10^{-2} |
| Crescent | (0.5, 0) | 2.07×10^{-1} | 6.28×10^{-2} | 3.62×10^{-2} |
| | (-0.5, 0) | 2.11×10^{-1} | 7.40×10^{-2} | 3.32×10^{-2} |
| | (0, 0.5) | 2.11×10^{-1} | 7.40×10^{-2} | 3.32×10^{-2} |
| | (0, -0.5) | 2.07×10^{-1} | 6.28×10^{-2} | 3.62×10^{-2} |

Table 4: Diagonal Translation Errors

three methods, however, [SLIC](#) and [FLAP](#) suffered the most from this which can be seen clearly in figure [46](#).

3.2.2 Rotation

The second test determined the method's ability to track rotational movement. A rotational velocity field was applied to the hollow rotated square used in the translational tests and a slotted circle, a standard test object for advection algorithms and volume tracking methods. The test objects were placed both with the shape centre collocating to the rotational centre and where the shape is offset, such that the shape edge is located at the centre of rotation. The equations used to generate the required velocity field are as follows

$$\begin{aligned}
 \delta x^* &= \frac{1}{n_{\text{cols}} - 2}, & \delta y^* &= \frac{1}{n_{\text{rows}} - 2}, \\
 x &= (i + 0.5)\delta x^*, & y &= (j + 0.5)\delta y^*, \\
 u &= 0.5 - y, & v &= x - 0.5.
 \end{aligned}$$

The result of one full rotation of the collocated slotted disk and hollow square can be seen in figure [47](#). The errors generated by the methods in rotational testing are documented in table [5](#). The results of the rotational test once again show the Youngs' method consistently maintains the object shape to the highest degree of fidelity. Interestingly, the [SLIC](#) method returned a better result than the [FLAP](#) method for the slotted disk centred about the centre of rotation, however, produced a much worse result for the hollow square. Once again the [SLIC](#)

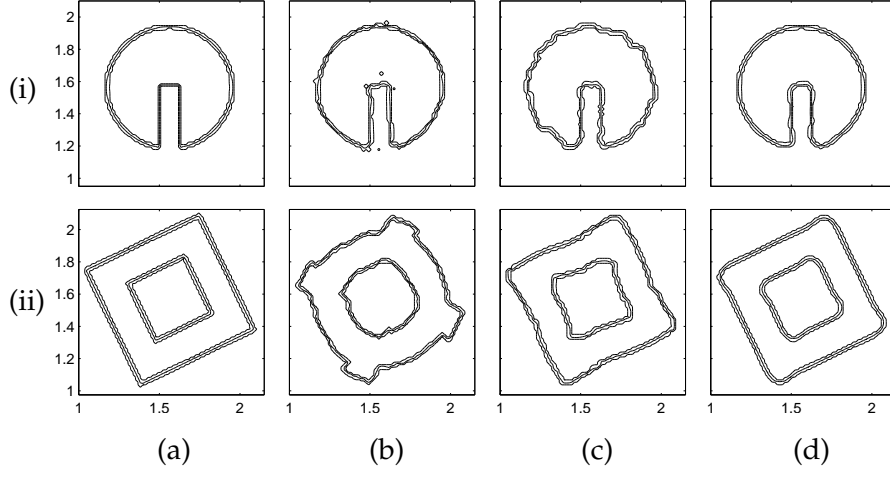


Figure 47: Result after one full rotation, for (i) a Zalesak slotted disk test object and (ii) a hollow rotated square test object, where the centre of the test object is collocated with the centre of the velocity field. (a) Initial conditions, (b) *SLIC* (c) *FLAP* and (d) *YVOF*.

| Shape | Location | <i>SLIC</i> | <i>FLAP</i> | <i>YVOF</i> |
|---------------|----------|-----------------------|-----------------------|-----------------------|
| Zalesak Disk | centre | 3.32×10^{-2} | 4.64×10^{-2} | 1.02×10^{-2} |
| | offset | 6.41×10^{-2} | 3.60×10^{-2} | 1.69×10^{-2} |
| Hollow Square | centre | 1.44×10^{-1} | 6.81×10^{-2} | 2.95×10^{-2} |
| | offset | 1.25×10^{-1} | 5.19×10^{-2} | 3.05×10^{-2} |

Table 5: Rotational Errors

method caused the appearance of some flotsam and jetsam artefacts in the slotted disk test.

3.2.3 Shear

The third and final test studied how well the methods handled shearing flow conditions. The velocity field was defined by

$$\begin{aligned}\delta x^* &= \frac{\pi}{\text{ncols} - 2} & \delta y^* &= \frac{\pi}{\text{nrows} - 2} \\ x &= (i + 0.5)\delta x^* - \frac{\pi}{2} & y &= (j + 0.5)\delta y^* - \frac{\pi}{2} \\ u &= \cos(x) \sin(y) & v &= -\sin(x) \cos(y)\end{aligned}$$

The test was run with the above velocity conditions for the first half of the test, after which the sign of the velocity conditions was reversed for the remaining half of the test, where m is defined as the total number of steps. The fluid, having returned to its original position, can then be compared to the initial conditions. The shear test was conducted with a circle (diameter $\pi/2.5$), rotated square (width $\pi/2.5$, angle 0.45°), hollow circle (external diameter $\pi/2.5$, internal diameter $\pi/5$) and hollow square (external width $\pi/2.5$, internal width $\pi/5$). The errors for the shearing tests can be found in table 6.

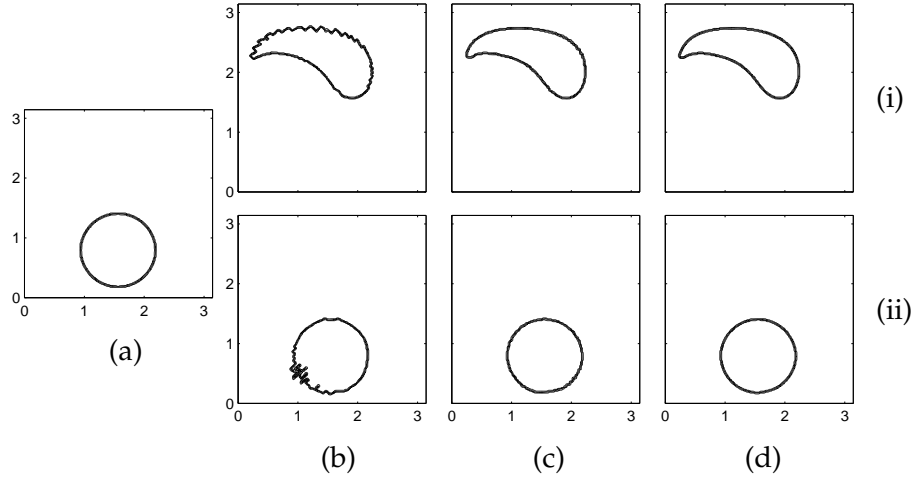


Figure 48: Results of shearing test after (i) 500 steps in one direction and (ii) 500 steps in the other direction. (a) Initial conditions, (b) [SLIC](#) (c) [FLAP](#) and (d) [YVOF](#)

The shear flow test showed both the Youngs' method and the [FLAP](#) method to reproduce the initial conditions with a high degree of fidelity, with the Youngs' method generally producing a narrowly better result. Both methods outperformed the [SLIC](#) method which did quite poorly in this test.

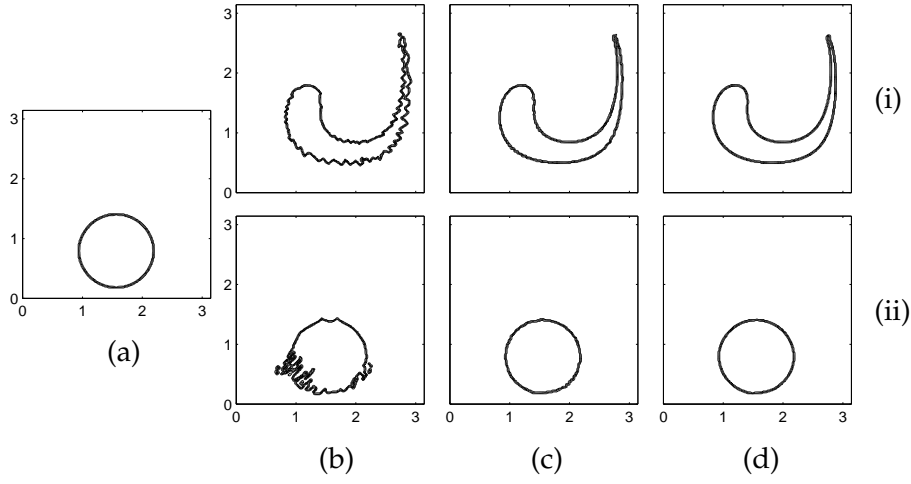


Figure 49: Results of shearing test after (i) 1000 steps in one direction and (ii) 1000 steps in the other direction. (a) Initial conditions, (b) [SLIC](#) (c) [FLAP](#) and (d) [YVOF](#)

| Shape | m | SLIC | FLAP | YVOF |
|---------------|------|-----------------------|-----------------------|-----------------------|
| Circle | 1000 | 5.0×10^{-2} | 2.34×10^{-2} | 2.14×10^{-2} |
| | 2000 | 1.19×10^{-1} | 2.36×10^{-2} | 2.15×10^{-2} |
| | 3000 | 1.77×10^{-1} | 2.47×10^{-2} | 2.07×10^{-2} |
| | 4000 | 2.24×10^{-1} | 2.39×10^{-2} | 2.53×10^{-2} |
| Hollow Circle | 1000 | 1.07×10^{-1} | 4.88×10^{-2} | 4.26×10^{-2} |
| | 2000 | 2.04×10^{-1} | 4.83×10^{-2} | 4.36×10^{-2} |
| | 3000 | 2.97×10^{-1} | 5.16×10^{-2} | 4.30×10^{-2} |
| | 4000 | 4.08×10^{-1} | 5.40×10^{-2} | 4.97×10^{-2} |
| Square | 1000 | 5.16×10^{-2} | 2.17×10^{-2} | 1.72×10^{-2} |
| | 2000 | 9.96×10^{-2} | 2.44×10^{-2} | 1.66×10^{-2} |
| | 3000 | 1.22×10^{-1} | 2.58×10^{-2} | 1.95×10^{-2} |
| | 4000 | 1.45×10^{-1} | 3.04×10^{-2} | 2.21×10^{-2} |
| Hollow Square | 1000 | 8.99×10^{-2} | 4.26×10^{-2} | 3.42×10^{-2} |
| | 2000 | 1.64×10^{-1} | 4.60×10^{-2} | 3.39×10^{-2} |
| | 3000 | 1.95×10^{-1} | 5.29×10^{-2} | 3.80×10^{-2} |
| | 4000 | 2.25×10^{-1} | 6.21×10^{-2} | 4.47×10^{-2} |

Table 6: Shearing Errors

3.3 SEDIMENT TRANSPORT TESTS

Unlike the fluid tracking and advection methods tested above, the sediment advection does not require the calculation of an interface as the assumption is made that the concentration of sediment is evenly distributed in the volume of fluid present in the cell. The sediment transport model was tested by applying a sediment settling test, an angle of repose test and a constant velocity, equilibrium profile test.

3.3.1 *Settling*

The first test case sought to validate settling and deposition of material. This was conducted by placing a uniform concentration in the filled control volume with zero fluid velocity. Knowing the dimensions of the test area and the settling velocity of the test sediment, the theoretical time for all material to have settled can be calculated as the time taken for a sediment particle to fall the entire height of the control volume. The simulation was conducted with a condition allowing sediment concentration near the bed to be converted into an increased bed height, the normal procedure for the model. The simulation was also conducted with a condition that dictated that sediment was not to be converted into increased bed height, causing the concentration to accumulate in the lowest row. The results of both simulations can be seen in figure 50.

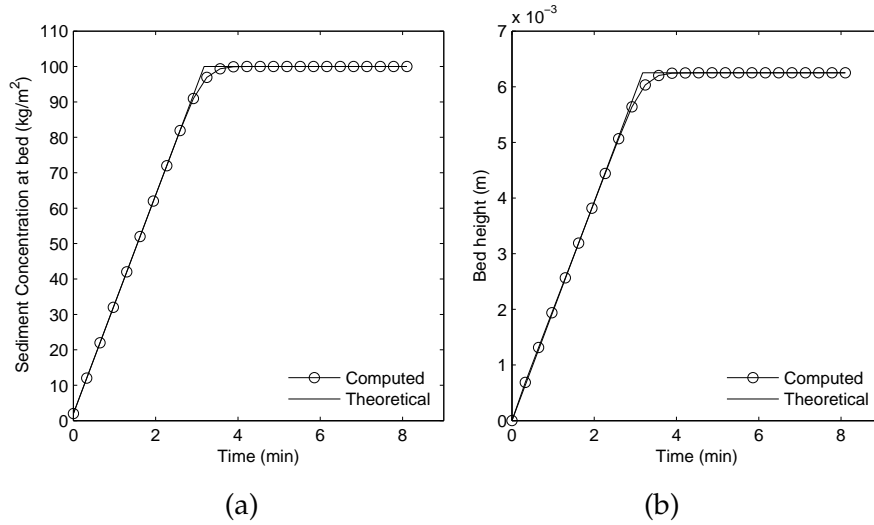


Figure 50: Results of the settling test for (a) a test without transition, where concentrations near the bed are not converted into increased bed height and (b) a test with transition where concentrations near the bed are converted into increased bed height.

The result of both simulations shows good agreement with the theoretical result. The simulation where bed transition was allowed, pro-

duced slightly divergent results from the theoretical line however, the theoretical line does not take into account effect of the raising bed.

3.3.2 *Angle of repose*

The critical angle of repose redistribution algorithm, or slope limiter was tested by applying a variety of simple initial bed states and allowing the bed to evolve. A grid size of 202×202 (200×200 real domain cells) was used on a domain with a height and width equal to one. The error was judged by conservation of total bed height. The equation for the error is

$$E = \frac{\left\| \sum_{i,j=1}^{i_{\max},j_{\max}} h_{i,j}^n - \sum_{i,j=1}^{i_{\max},j_{\max}} h_{i,j}^0 \right\|}{\sum X_{i,j}^0} \quad (104)$$

Figure 51 shows the results for bed discontinuities placed at either side of the domain, for a discontinuity of height 0.1, 0.5 and 0.9. The results show that the algorithm successfully reconstructs discontinuities using the trough and the peak method, for cases where the new triangle (a) entirely falls within the domain, (b) is adjacent to one domain edge and (c) is adjacent to both domain edges. Table 7 contains the errors associated with the plateau tests and the errors all fall within the range of computational floating point error.

Figure 52 shows the results for positive and negative bed discontinuities placed centrally within the domain, for a discontinuity of height 0.1, 0.5 and 0.9. Table 7 contains the errors associated with the plateau tests. Figure 52 clearly shows a growing asymmetry with increasing discontinuity height. This is because the algorithm, for discontinuities, will sweep across summing the area until the area will create a new triangle height that would breach the bed surface. In the case of larger discontinuities the new triangle height is much smaller than the original height of the discontinuity and as such the entire area is smoothed in a single direction. This creates a triangle and a new discontinuity, which is smoothed in the other direction, however much of the sediment has already been smoothed further towards the initial direction. The errors show no increased loss in bed height and as such it can be concluded that this result represents correct functioning order for the algorithm provided. It has been shown, however, that the algorithm will produce non realistic, increasingly asymmetric results for large discontinuities.

Figure 53 shows the results for peaks and troughs with slope angles greater than the critical angle of repose. Three bed configurations are shown, a single peak or trough, two peaks or troughs of equal height and three peaks or troughs where the central peak or trough is of greater height than the peaks/troughs on either side. The width of the overall feature is equal to the width of the domain. Table 8 and table 9 show the errors in conservation of sediment volume for

the peaks and troughs respectively. In addition to the wide results shown in figure 53, tables 8 and 9 also include the errors produced by the same configurations where the features have a width of one half of the domain width. All errors lie within the range of computational floating point error. The results in figure 53 show that the algorithm successfully reconstructed the bed formations. It must be noted that the more complex geometry of the triple peak required multiple smoothing steps to find a stable bed formation. In addition, the final bed formation of the triple peak has some slight asymmetries, however, this will always be the case when slopes are smoothed in order from left to right. These asymmetries are very slight compared to those produced by the large, centrally located discontinuity tests. Future revisions of the algorithm should look to impose different slope orders, however, any order is likely to cause some asymmetry as they are an approximation to a process that occurs concurrently for all slopes.

| Plateau location | Method | 0.1 | 0.5 | 0.9 |
|------------------|--------|----------------------|----------------------|----------------------|
| Left | trough | 7.8×10^{-8} | 9.1×10^{-9} | 1.4×10^{-8} |
| Right | peak | 4.7×10^{-8} | 1.6×10^{-8} | 1.4×10^{-8} |
| Centre | trough | 1.1×10^{-7} | 1.0×10^{-7} | 4.6×10^{-8} |
| | peak | 9.6×10^{-8} | 1.4×10^{-7} | 8.6×10^{-8} |

Table 7: Repose test errors, plateaus, where 0.1, 0.5 and 0.9 are the initial plateau heights above the bed

| I.C. | Peak Type | Single | Dual | Triple |
|--------|-----------|----------------------|----------------------|----------------------|
| narrow | point | 1.3×10^{-7} | 3.0×10^{-7} | 2.3×10^{-7} |
| | even | 2.7×10^{-7} | 9.8×10^{-8} | 2.5×10^{-7} |
| | odd | 2.2×10^{-7} | 2.3×10^{-8} | 1.0×10^{-7} |
| wide | point | 3.8×10^{-8} | 1.5×10^{-8} | 5.8×10^{-8} |
| | even | 4.0×10^{-8} | 1.5×10^{-7} | 2.6×10^{-8} |
| | odd | 3.8×10^{-8} | 1.0×10^{-8} | 2.1×10^{-7} |

Table 8: Repose test errors, peaks

3.3.3 Equilibrium conditions

The third test used to validate the sediment transport model looked at generating the concentration profile of a flow in equilibrium conditions. This test applied a steady, uniform fluid flow with either a uniform concentration or an equivalent mass of sediment placed at

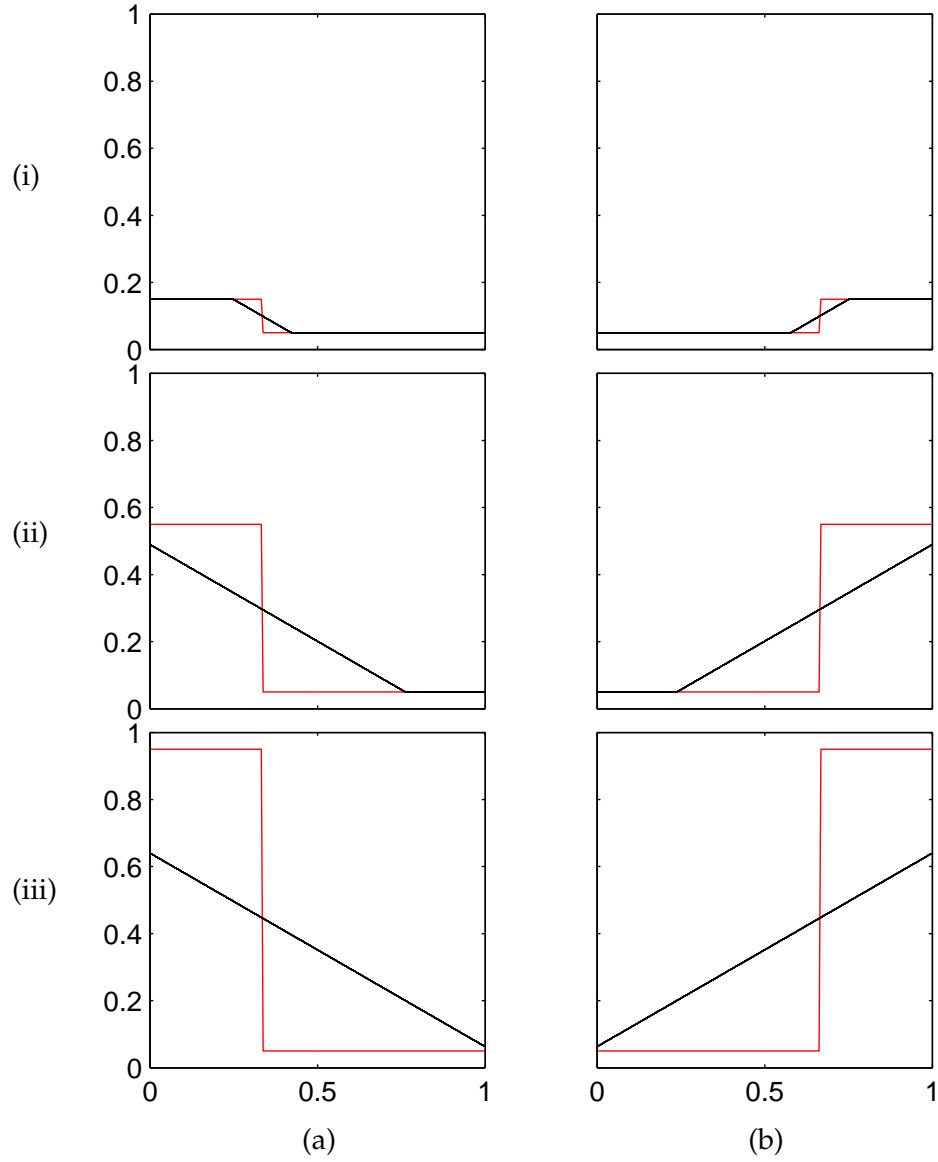


Figure 51: Results of the repose test on a 202×202 grid for a plateau, one third of the width of the domain, placed against (a) the left hand or (b) the right hand edge of the domain. Initial plateau height is (i) 0.1, (ii) 0.5 and (iii) 0.9 above the bed. Initial conditions shown in red.

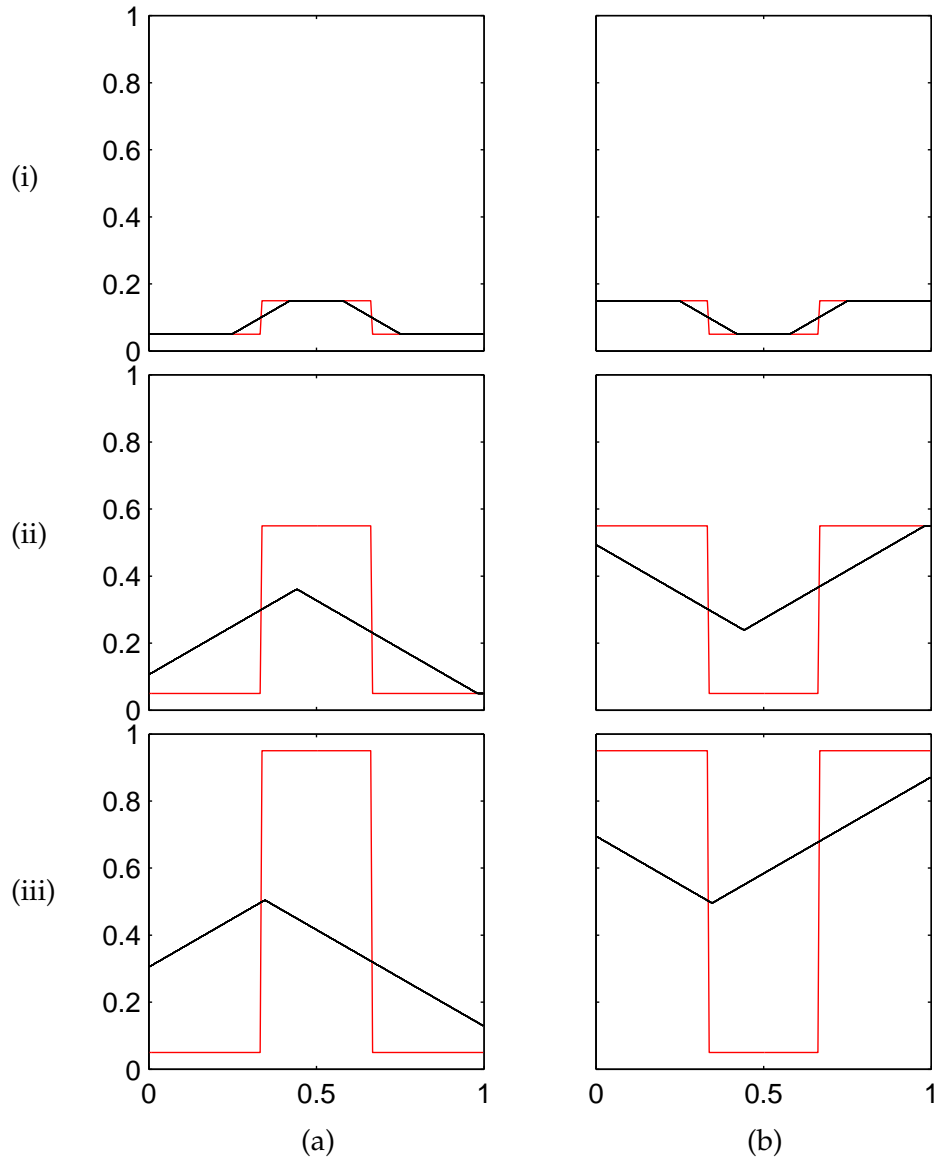


Figure 52: Results of the repose test on a 202×202 grid for a centrally placed plateau, one third of the width of the domain. Plateau exists as either (a) a peak or (b) trough. Initial plateau height is (i) 0.1, (ii) 0.5 and (iii) 0.9 above or below the bed. Initial conditions shown in red.

| I.C. | Peak Type | Single | Dual | Triple |
|--------|-----------|----------------------|----------------------|----------------------|
| narrow | point | 1.1×10^{-8} | 8.1×10^{-8} | 6.6×10^{-8} |
| | even | 6.4×10^{-8} | 4.6×10^{-8} | 1.1×10^{-7} |
| | odd | 3.2×10^{-8} | 1.1×10^{-8} | 2.2×10^{-8} |
| wide | point | 3.3×10^{-8} | 2.5×10^{-7} | 3.9×10^{-8} |
| | even | 3.5×10^{-8} | 2.5×10^{-7} | 6.4×10^{-8} |
| | odd | 1.7×10^{-8} | 1.8×10^{-8} | 9.2×10^{-7} |

Table 9: Repose test errors, troughs

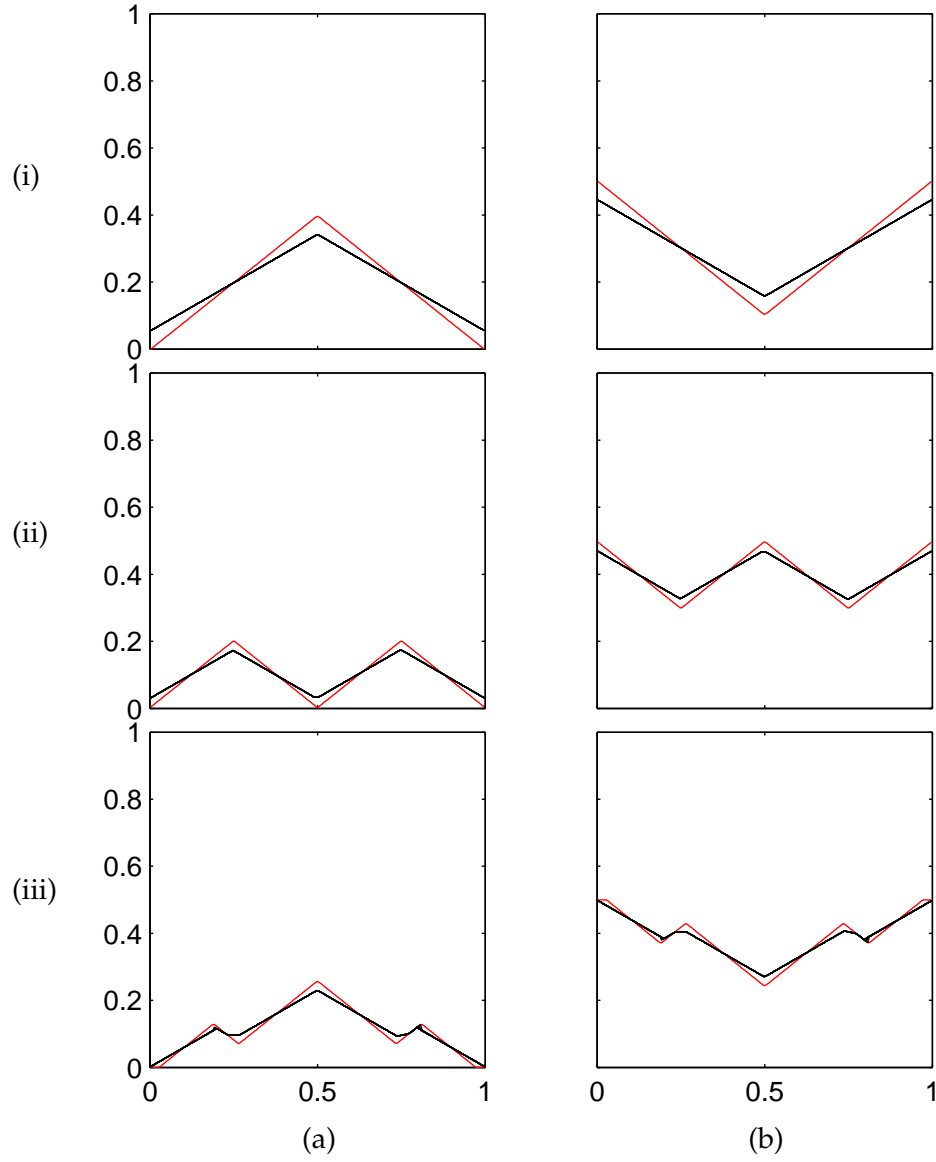


Figure 53: Results of the repose test on a 202×202 grid for (a) a peak or (b) a trough in a (i) single, (ii) dual or (iii) triple configuration. All initial slopes are greater than the critical angle of repose and the feature width is equal to the width of the domain. Initial conditions shown in red.

the bed. For steady, uniform flow, the advection-diffusion equation simplifies to

$$Cv_s + Ky \frac{dC}{dy} = 0, \quad (105)$$

for which a solution is the Rouse profile, given by equation

$$\frac{C}{C_a} = \left(\frac{a(d_f - y)}{y(d_f - a)} \right)^\lambda, \quad (106)$$

where a is the reference height and C_a is the reference concentration, the concentration at height a above the bed and λ is the Rouse number (settling velocity divided by the product of the Von Karman constant and the shear velocity) divided by the β -factor, $\lambda = v_s/(\beta\kappa u_*)$. The reference concentration is taken to equal the concentration in the first cell above the bed for the given width, and the reference height is the height of the midpoint of the cell above the bed.

In order to test the sediment transport model, the fluid damping factor, ϕ , was initially set to $\phi = 1$. The ϕ factor is commonly used as a curve fitting factor in empirical studies [Van Rijn \[1984b\]](#). Tests were conducted on a range of sediment diameters, flow velocities and concentrations. Firstly, the development of the profile was analysed for the uniform and equivalent bed concentration initial conditions. The sediment profiles during development differ, however, the profile at equilibrium match exactly. Figures 54 and 55 show the development of the equilibrium profiles from the two different initial conditions and figure 56 shows the exact collocation of the equilibrium states. In addition to this, figure 56 also shows the exact collocation of results produced with different sediment concentrations for simulations run with a ϕ factor of 1. The use of a ϕ factor that is a function of the concentration as suggested by [Van Rijn \[1984b\]](#) will cause a change in the concentration profile and the results will no longer collocate on non-dimensional plots. Changing the sediment diameter was seen to significantly change the sediment profile, with reduction in diameter causing a shift towards a more uniform distribution of suspended material, see figure 57. The difference between the computed equilibrium profiles and the equivalent Rouse profile can be seen to increase with increasing sediment diameter.

Choosing the flow velocity also significantly changed the sediment profiles, with increase in flow speed causing a shift towards a more uniform distribution of suspended material, see figure 58. The difference between the computed equilibrium profiles and the equivalent Rouse profile can be seen to decrease with increasing flow velocity.

By choosing various sediment particle diameter and velocity conditions to produce specific λ values (specifically $\lambda = 0.1, 0.3$ and 0.5), the non-dimensionalised results could be plotted together showing that the equilibrium profiles match exactly for identical λ values, see

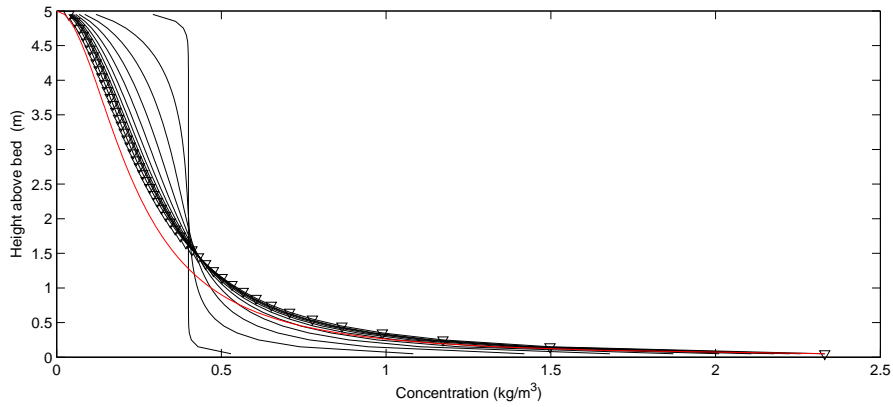


Figure 54: Developing (black lines) and equilibrium profile (triangular markers) developed from a uniform initial condition, $d = 154$, $U = 2$. Rouse profile shown in red.

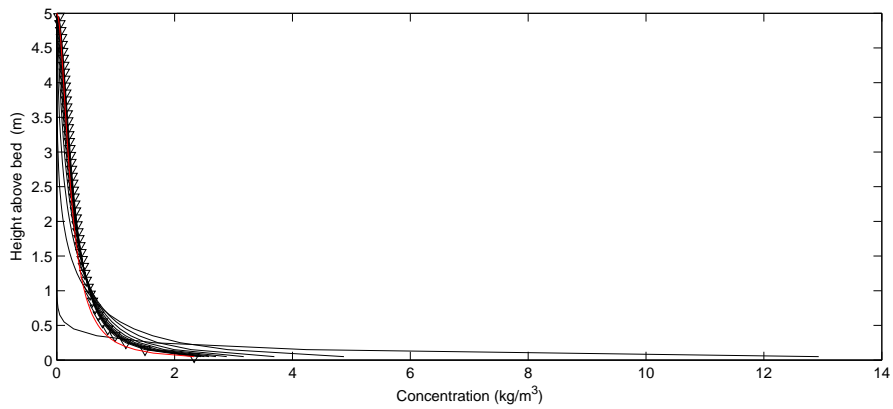


Figure 55: Developing (black lines) and equilibrium profile (triangular markers) developed from a sediment at bed initial condition, $d = 154$, $U = 2$. Rouse profile shown in red.

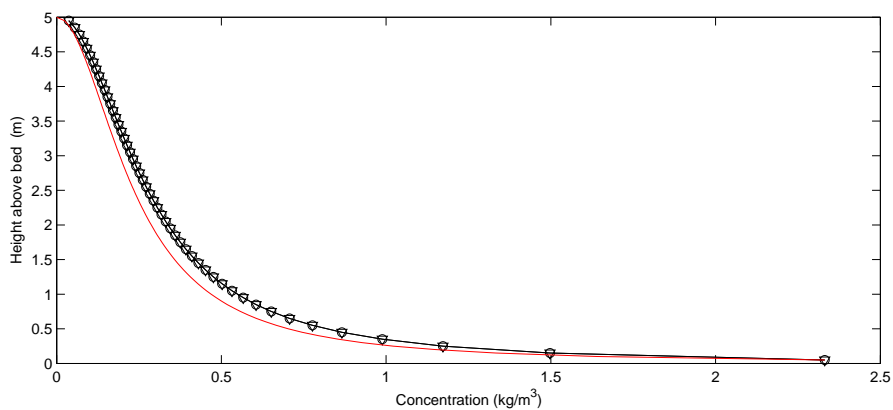


Figure 56: Comparison of equilibrium profile developed from a uniform initial condition (triangular markers) and a sediment at bed initial condition (circular markers), $d = 154$, $U = 2$. Rouse profile shown in red.

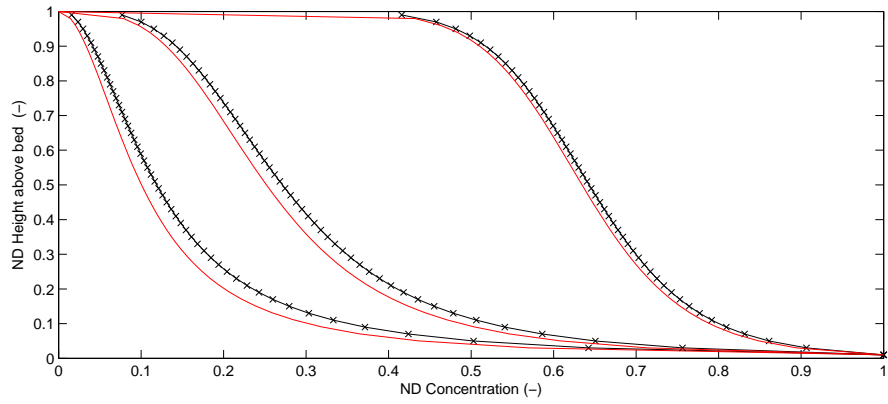


Figure 57: Equilibrium results for varying sediment diameters with a constant flow velocity, $U = 2$. $d = 154, 111, 57 \mu\text{m}$ from left to right. Rouse profiles shown in red.

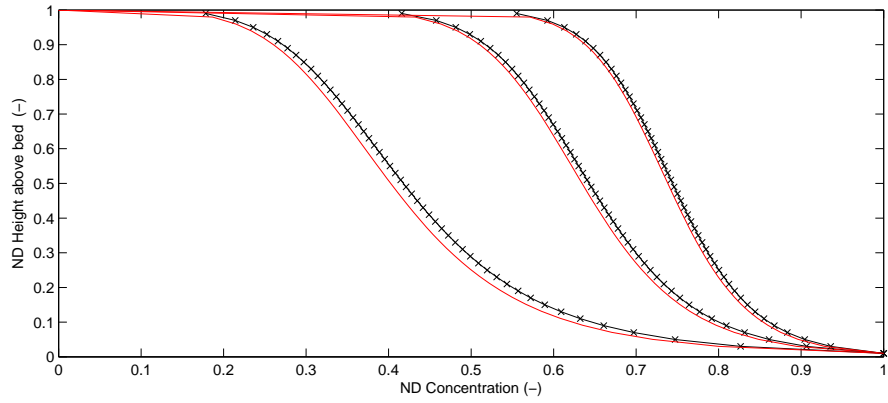


Figure 58: Equilibrium results for varying flow velocities with a constant sediment diameter, $d = 57 \mu\text{m}$. $U = 1, 2, 3$ from left to right. Rouse profiles shown in red.

figure 59. The difference between the computed equilibrium profiles and the equivalent Rouse profile can be seen to increase with increasing values of λ , which is consistent with the results in figure 57, as increasing sediment diameter will increase the value of λ . This is also consistent with the results in figure 58, as increasing flow velocity will decrease the value of λ .

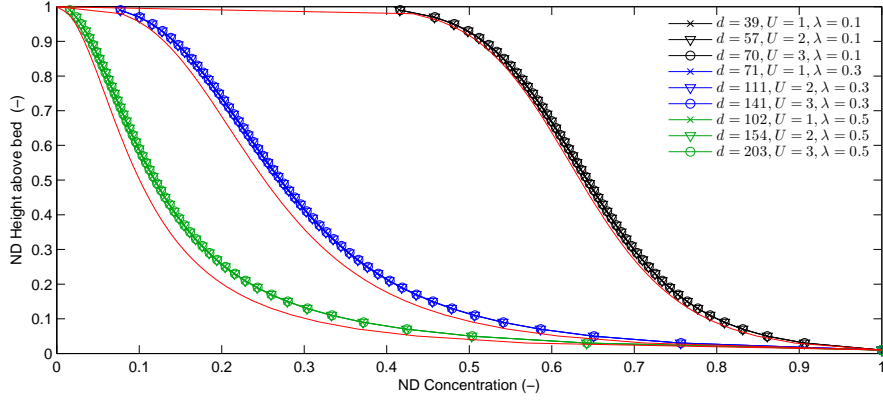


Figure 59: Equilibrium results for three different sediment diameter and velocities producing a λ value of 0.1, 0.3 and 0.5. Respective Rouse profiles are shown in red.

The error between the computed profiles and the analytical profile was computed by summing the difference between the values at each height and dividing it by the sum of either the analytical or computed values. Generally, when stating this type of error they are given as a proportion of the exact solution, however in this case it is more useful to quote the error in terms of a proportion of the computed concentration sum as this allows for a factor to be calculated to improve the computational model and reduce the error. The errors are plotted in figure 60.

It was found that the error increased linearly with λ values between 0 and 0.5, tailing off there after. For the $0 < \lambda \leq 0.5$ region a factor of $\phi_l = 1 - 0.5\lambda$ was applied to the calculation of the vertical diffusivity, which significantly improved the results. A cubic interpolation allowed for a closer description of the error curve and thus produced the following factor

$$\phi_c = 1 - (-0.2912\lambda^3 + 0.1897\lambda^2 + 0.4343\lambda + 0.0165). \quad (107)$$

The cubic factor closely matched the results of the linear factor for values of λ between 0 and 0.5, where the linear approximation showed good fit. The cubic factor improved the results for large λ however the errors continue to be much larger for these cases. Both the linear and cubic reduction factor were found to be reducing the vertical diffusivity too much for values of λ greater than 0.344. Thus an additional fit factor was generated from these results,

$$\phi_{c+} = \phi_c + \phi_c(-0.8226\lambda^3 + 1.2356\lambda^2 - 0.2992\lambda + 0.0316). \quad (108)$$

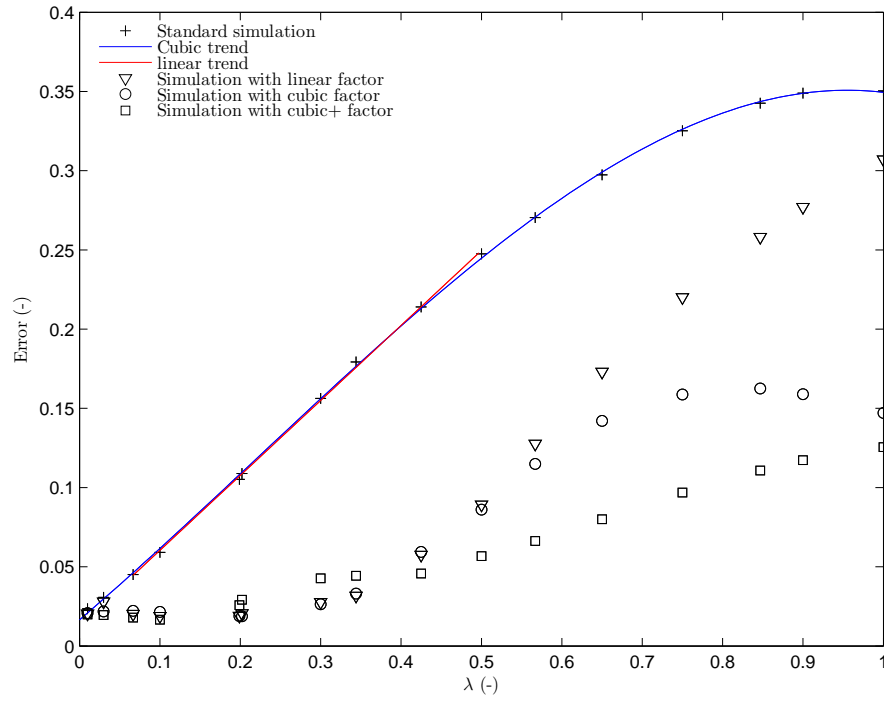


Figure 60: Equilibrium error results for various ϕ factors.

The results of the application of these factors are shown in figure 61, compared to the no factor case ($\phi = 1$) and the resulting errors are plotted in figure 60. The two left most curves ($\lambda = 0.75$ and $\lambda = 0.5$) in figures 61(b) and 61(c) show the over reduction in vertical diffusivity, causing the shift in the curve towards greater sediment concentration near the bed.

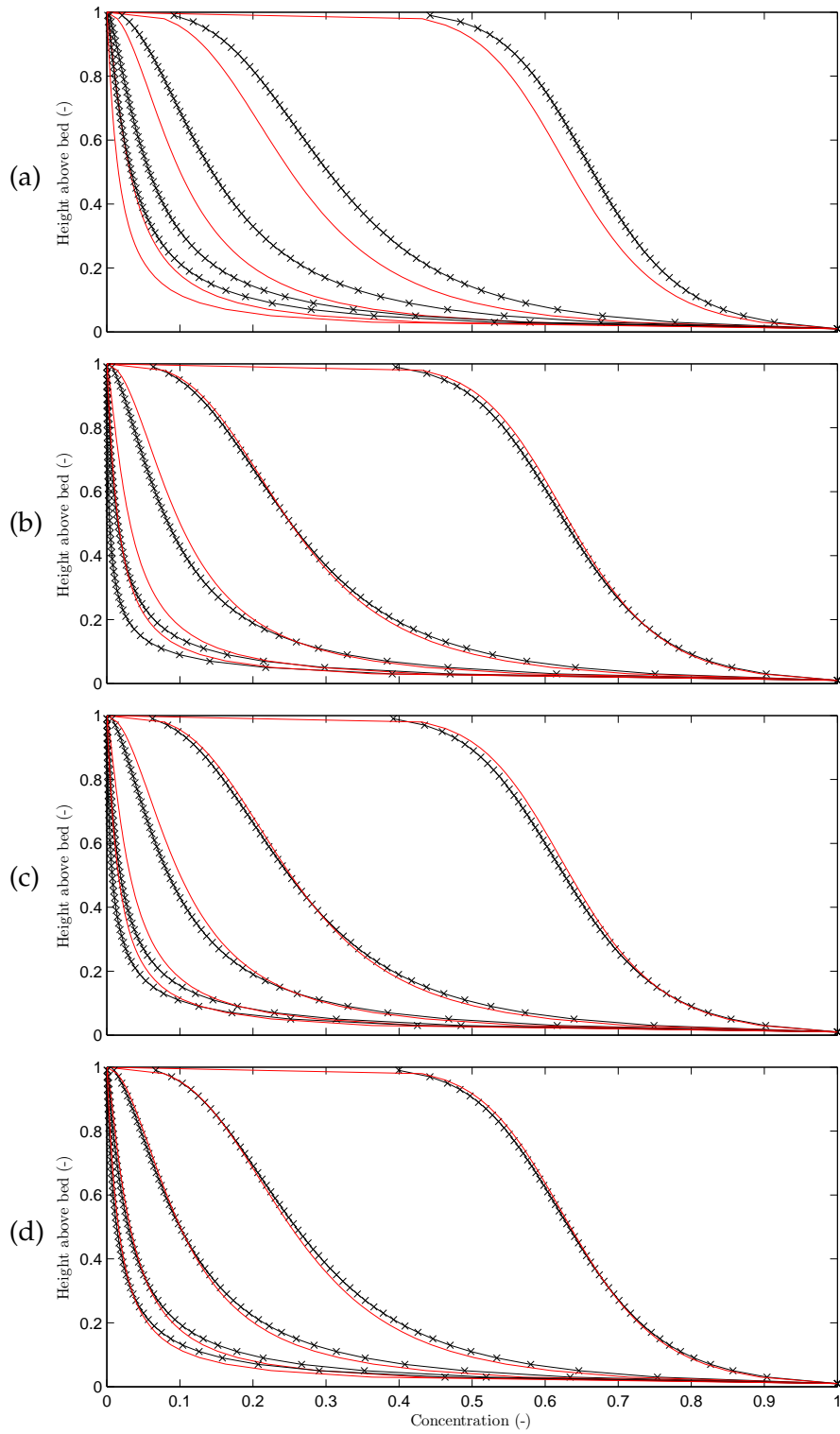


Figure 61: Equilibrium results for λ values of 0.75, 0.5, 0.3 and 0.1 from left to right. Figure (a) shows the results with no ϕ factor applied, figure (b) shows the results with the linear ϕ_l factor applied, figure (c) shows the results with the cubic ϕ_c factor applied and figure (d) shows the results with the “cubic plus” ϕ_{c+} factor applied,

DAM BREAK CASE STUDY

In this chapter, the numerical model described in chapter 2 is used to simulate the initial stage of flows that occur upon the release of a retained fluid of finite size from a behind retaining barrier. Various upstream and downstream depth conditions are tested for the rigid and erodible bed cases. These scenarios were also reconstructed in a series of laboratory experiments, to which the results of the simulation are compared. The experiments were designed to allow for validation of the model, but also to test the application of new experimental methods (new release mechanism) and data acquisition techniques (3D Particle Image Velocimetry) to experimental reconstruction of dam break flow.

4.1 EXPERIMENTAL APPARATUS AND PROCEDURE

The experiment used a small gated flume (with the dimensions $0.5 \times 0.2 \times 0.75\text{m}$, length \times height \times width) to retain a given volume of water, which would be allowed to collapse forward upon the vertical removal of the gate. The flume was precisely designed and constructed such that the gate will maintain a watertight seal as the pressure of the water behind the gate keeps it flat against the grooves in which it sits. This allows for a smooth and rapid removal of the gate, such that the removal was as close to the instantaneous removal assumed by the theoretical case as possible. The barrier was made from a thin 1mm thick stainless steel sheet so as to minimize the volume displaced by the barrier within the bed.

In addition to testing with a simple vertically removed gate, a second gate removal method was also designed to attempt to reduce the amount of disturbance and vorticity induced in the fluid as the gate is removed. The present mechanism is functionally identical by the barrier removal system designed by Dalziel [1993] in the study of Rayleigh-Taylor instability, however flume specific apparatus was required to provide an anchoring point. Dalziel required a barrier that would separate two fluids of different densities, where the more dense fluid laid on top of the less dense fluid until the barrier was removed, allowing the two fluids to mix. The normal removal of a barrier creates a strong vortex sheet on the face of the static fluids on either side of the barrier.

Experimental reconstruction of dam break suffers from the same problem of having shear induced vortical disturbances applied to the fluid immediately behind the barrier, as it is lifted. The aim of the

experimental reconstruction of the dam break problem is to produce a flow condition as close to the ideal conditions that are assumed in the theoretical analyses. This requires there to be as little disturbance to the liquid as possible, and it requires the release to be as fast as possible.

[Dalziel](#) created a system where in a flexible sheet, or fabric, was tethered at one end and wrapped around the end of the barrier such that the application of a force to the fabric would indirectly cause the removal of the barrier.

A system was designed in which a flexible sheet or piece of fabric was attached to the open top of the flume and wrapped underneath the gate. By pulling vertically upwards on the flexible sheet, the gate is indirectly moved upwards. The advantage of such a system is that the barrier will theoretically not impose a shear force on the retained fluid as the motion of the fabric is firstly perpendicular to the fluid face before vertically moving upwards on the side of the gate that is not in contact with the fluid. Some disturbance might still be induced by the curve produced by a flexible fabric moving around a sharp edge. The aim of the removal methods is to attempt to reconstruct the idealised release by minimizing both the disturbance to the fluid and time taken to fully remove the barrier. In order to capture flows that most closely match the idealised case for each method, both release mechanisms were tested using the maximum possible removal velocity. The wrapped-fabric removal method is illustrated in figure 62. The experimental apparatus was originally designed and created as part of a Masters project [[McMullin, 2010](#)]

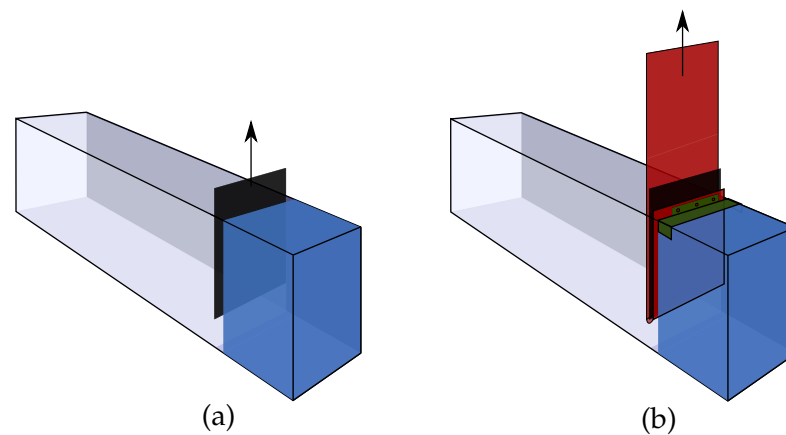


Figure 62: barrier removal mechanisms. (a) Simple vertical removal method, (b) Wrapped-fabric barrier removal method

Measurements were taken photographically, through a two camera system (two identical Nanosense Mk III cameras with Nikkor 60mm f2.8 lenses, attached via a Tilt shift squeeze box, mounted on a rail), so as to be able to use three dimensional Particle Image Velocimetry

(3D PIV) to capture the velocity field, not only in the laser-plane, but also perpendicular to it. Particle Image Velocimetry uses photographs of neutrally buoyant particles ($150\mu\text{m}$ Pliolite with a specific gravity of approximately 1.02g cm^{-3}), dispersed throughout a fluid, to track the motion of the fluid by analysing the motions required to move the particles from one image to the next. This is conducted in a single plane by using a narrow light sheet, or in this case, a laser sheet (provided by a Dantec DualPower 50-100 Neodymium-doped Yttrium Aluminium Garnet crystal (Nd-YAG) pulsed laser with a maximum pulse frequency of 100Hz, maximum output of 400mJ, pulse length of 4ns and wavelength between 106-532nm. The Laser beam is passed through a cylindrical optic to produce a 2-4mm wide sheet), to illuminate only the particles in the plane that is to be considered [Adrian and Westerweel, 2011]. Three dimensional (also known as Stereoscopic) PIV takes this further by placing two cameras at an angle to the plane, to photographically capture the particles which move away from or into the illuminated plane (see Prasad [2000]). Although the experiment is designed to produce results that are two dimensional in nature, the three dimensional capturing procedure allows the assumption of two dimensionality to be tested. See figure 63 for an illustration of the experimental set-up. Note that the maximum pulse frequency (and therefore exposure rate) of 100Hz laser was on the lower limit of what was required for the PIV software (Dynamic Studio v3.1) to resolve the velocity field. This may be the reason that some experimental runs did not resolve. It would be recommended for future work to use a higher frequency laser or some other lighting system.

Experiments were first conducted over a dry bed. A range of backwater or reservoir depths was tested from 0.06m to 0.14m with the average tested backwater height equal to 0.09m. Various tailwater heights were tested, ranging from 0.005m to 0.02m. The experimental equipment performed well for the rigid bed tests, however the introduction of bed sediment made it very difficult to maintain a watertight seal. Thus sediment tests were only successfully carried out with a non-zero tailwater condition above the sediment bed.

The sediment consisted of near-spherical $500\mu\text{m}$ Ballotini glass beads, with a known settling (or fall) velocity of 7.1cm/s, particle Reynolds number of 35.5, a repose angle of 26° and a relative density value of 2.5 [Munro et al., 2009].

The sediment was laid evenly at the base of the tank and levelled using a custom made rake designed to run across the top of the tank, levelling the sediment to exactly the same height for every run.

The experiment was conducted by placing the barrier within the flume, and filling the reservoir to a specific depth. Pliolite was prepared and added to the water and dispersed throughout. In the case of sediment bed runs, the sediment required levelling. The laser was

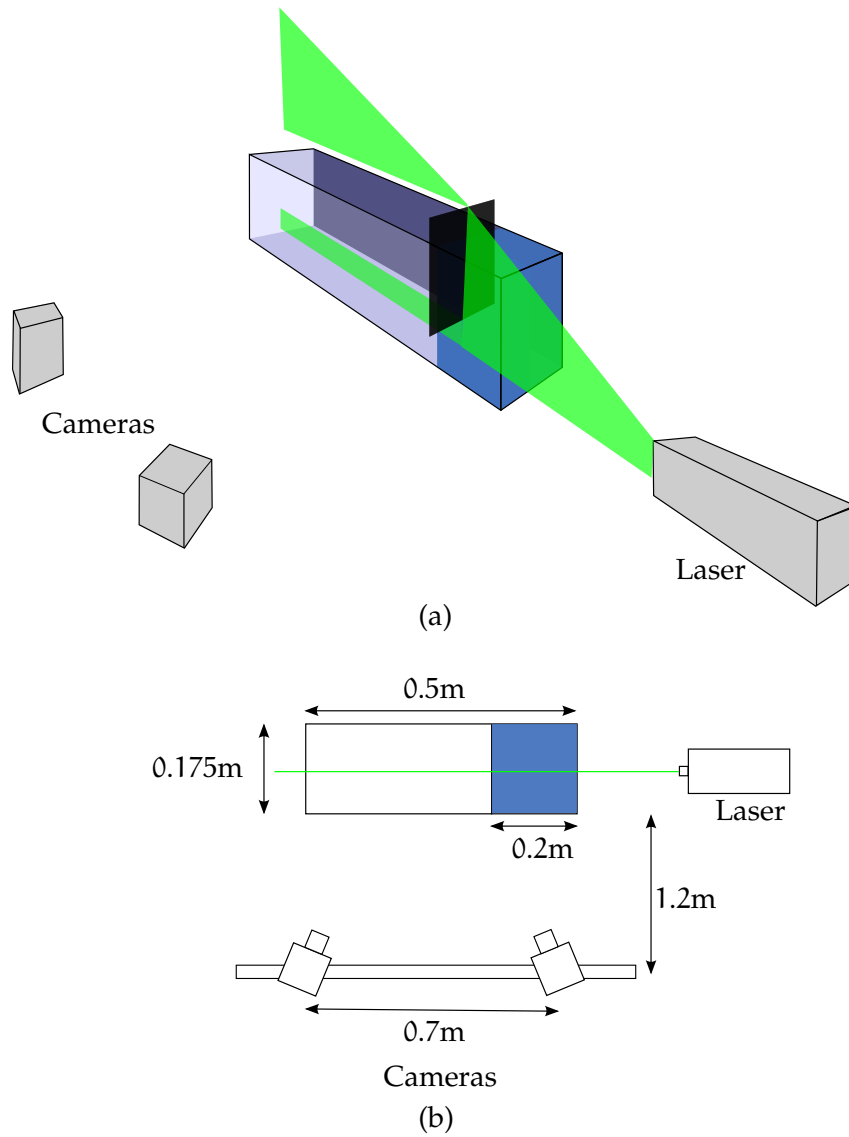


Figure 63: Experimental setup. (a) 3D view (shown during the removal of the barrier), (b) Plan view.

switched on and cameras captured the flow that occurred once the barrier was lifted. The captured images were then processed in order to apply the PIV software (Dynamic Studio v3.1).

Figure 64 illustrates the initial conditions used for the experiments and their respective numerical simulations.

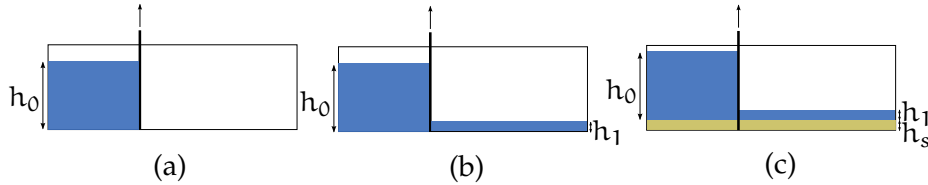


Figure 64: Initial conditions. (a) Dry rigid bed, (b) non-zero tailwater over a rigid bed, (c) non-zero tailwater over a granular bed.

4.1.1 Image Processing

Each captured image underwent a significant amount of processing so as to produce the most accurate velocity data. The first stage of the processing was the application of a hand drawn mask to every frame. The purpose of applying a mask is to remove any image data located outside the flow, which could confuse the PIV software. An image processing algorithm was developed to find the surface (so as to apply the mask outside of the surface), however inconsistencies of exposure required exact tuning that was more time intensive than the manual masking of each frame. The masking process ensured that nothing outside of the flow is taken into account by the PIV software, and also produced surface position plots for all frames. The second step is to apply arithmetic image manipulation, a subtraction to reduce all of the blacks down to pure black and a multiplication to increase the brightness of the remaining points, the lit Pliolite particles. These steps were also conducted for the images of calibration plates that were placed in the laser sheet so as to accurately determine the angle of the cameras to the laser sheet and use this information to 'dewarp' the images captured in the experimental run. Calibration was performed using the Direct Linear Transformation method given as IMF:DLT by Dynamic Studio. The results of the calibration from both cameras and the results of the preprocessing of the experimental images (masking, arithmetic manipulation, adaptive correlation, moving average validation and vector masking) are selected for 3D PIV analysis, producing the velocity field.

4.2 RESULTS

4.2.1 *Simple vs. Wrapped-fabric release mechanism*

Figure 65 shows a comparison of the captured photographs for a run produced by the simple and wrapped-fabric release mechanisms. The images in figure 65 are the negatives of the original images, which were shot in darkness. The black dots that can be seen in figure 65 are the Pliolite particles being illuminated in a single plane of the flow.

The difficulty in comparison of these results is in how $t = 0$ is defined for the two non instantaneous releases, where the release occurred over different time scales for each release mechanism. The increased release time is inherent to the wrapped fabric method as the fabric must move twice the distance vertically to remove the barrier, doubling the time taken to remove for a given removal rate applied to the simple barrier and the fabric.

Figure 65 shows the results for the same value of t where $t = 0$ at the initiation of motion of the barrier. Figure 66 compares the results for the same value of t where $t = 0$ is considered to be the point at which the barrier's lower edge recedes from the fluid. The front location of the two methods are comparable for the latter definition of t . The resulting surface profiles are largely similar, with some key differences. The dam break flow produced by the wrapped-fabric mechanism, flowed for a greater amount of time at the point of comparison and so the flow depth is greater beyond the dam site ($x > 0$) and smaller behind the dam site ($x < 0$). The contact angle between the tip and the rigid bed is almost 90° in both cases, however the angle made by the head and the rigid bed in the region behind the tip is steeper for the wrapped-fabric mechanism. This is probably also caused by the increased flow height ahead of the dam site.

Similar comparison can be made for the non-zero tailwater case. In this case, the difference in front position between the two methods is much less. Figures 67 and 68 show the comparison of dam break for the two definitions of $t = 0$. Neither of the two previous definitions apply well to this case and so an offset time between these definitions is required for continued comparison of front progression. Figures 67 and 68 confirm the result of Stansby et al. [1998], where dam break over a tailwater creates a mushroom-like protrusion in the surface profile for the initial stages. This goes on to form a very small breaking wave. The size of the breaking wave and the distance until the wave breaks was found to increase with increasing tailwater depths, up to a limit when the reducing height difference reduces the breaking wave characteristics.

Figures 69 and 70 show a comparison of results for the tailwater case over a sediment bed. In this case the second definition for $t = 0$

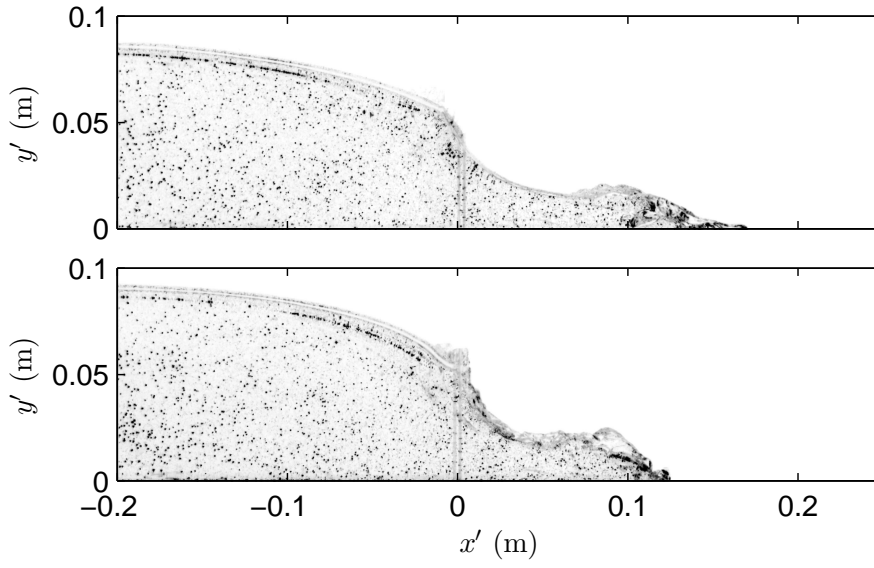


Figure 65: Comparison of photographic results for a dry rigid bed. (a) simple and (b) wrapped-fabric release mechanisms. $t' = 0.15$ s where $t' = 0$ is the initiation of barrier removal.

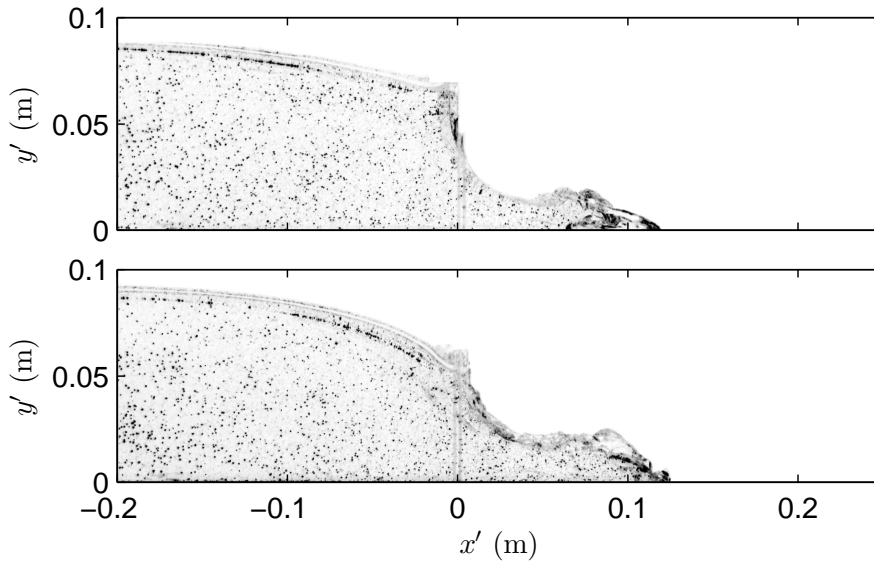


Figure 66: Comparison of photographic results for a dry bed. (a) simple and (b) wrapped-fabric release mechanisms, at $t = 0$ where $t = 0$ is the recession point of the lower edge from the fluid.

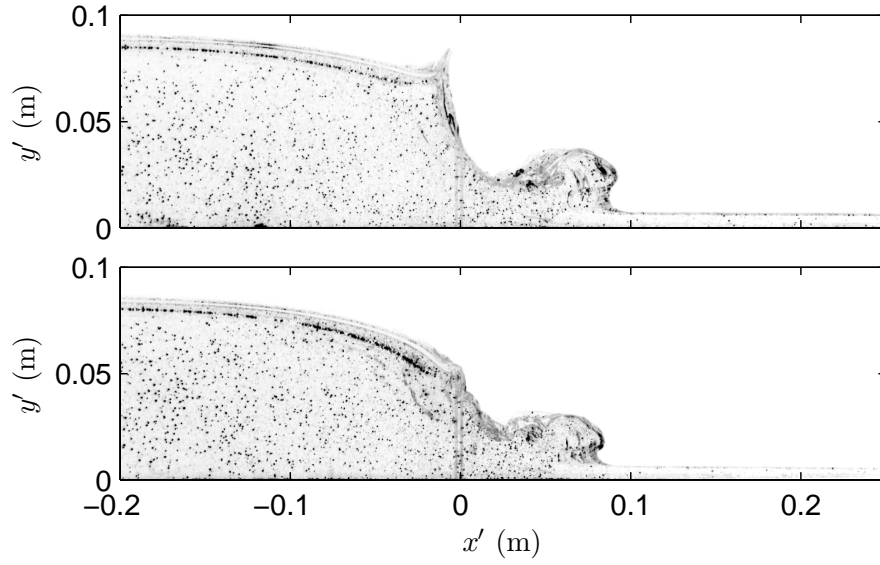


Figure 67: Comparison of photographic results for tailwater. (a) simple ($h_0 = 0.094\text{m}$ $h_1 = 0.0088\text{m}$) and (b) wrapped-fabric release mechanisms ($h_0 = 0.086\text{m}$ $h_1 = 0.007\text{m}$). $t' = 0.15\text{s}$ where $t' = 0$ is the initiation of barrier removal.

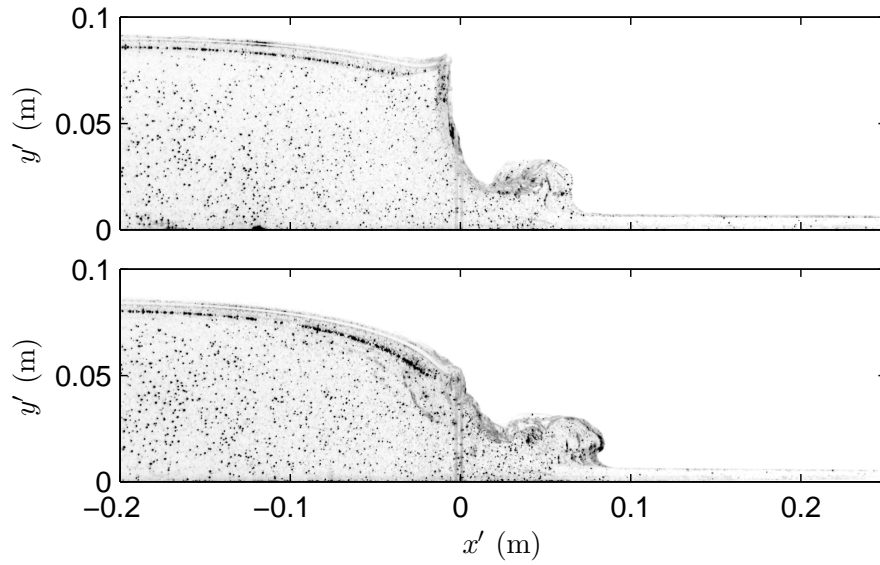


Figure 68: Comparison of photographic results for a tailwater over a rigid bed. (a) simple ($h_0 = 0.094\text{m}$ $h_1 = 0.0088\text{m}$) and (b) wrapped-fabric release mechanisms ($h_0 = 0.086\text{m}$ $h_1 = 0.007\text{m}$), at $t = 0$ where $t = 0$ is the recession point of the lower edge from the fluid.

(that $t = 0$ is defined at the point where the lower edge of the barrier recedes from the fluid) produces good agreement for front position between the two methods. This test was conducted with a larger tailwater height and the figures show the initial stages of a much larger

breaking wave that is forming. In addition a limited judgement can be made on the sediment motion. Experiments conducted using the simple removal method consistently suspended more sediment, which was observed by the exposure of the sediment as a single low-contrast entity, whereas the results produced by the wrapped-fabric mechanism showed illuminated sediment particles with a higher contrast to the surrounding water. In addition, the figures show that the sediment more thoroughly blocked the laser sheet, causing no Pliolite particles to be illuminated in the region ahead of the sediment for the simple release case, whereas some Pliolite particles were illuminated in this region in the wrapped-fabric release mechanism case. The final observation that indicates a greater amount of sediment suspension is larger reduction in bed height at the dam location for the simple removal case. This must only be considered partially indicative as it is not clear whether this is due to suspended load or bed load transport. The reduced suspended sediment load was observed for all runs over a sediment bed (with a tailwater) conducted with the wrapped barrier. It is not clear whether this is as a result of a reduction in the imposed vorticity by the shearing action of the vertical removal of the barrier, or whether this is as a result of the increased release time associated with the method. The latter may be true as the release is more gradual, potentially reducing the velocities involved in the suspension of the sediment material.

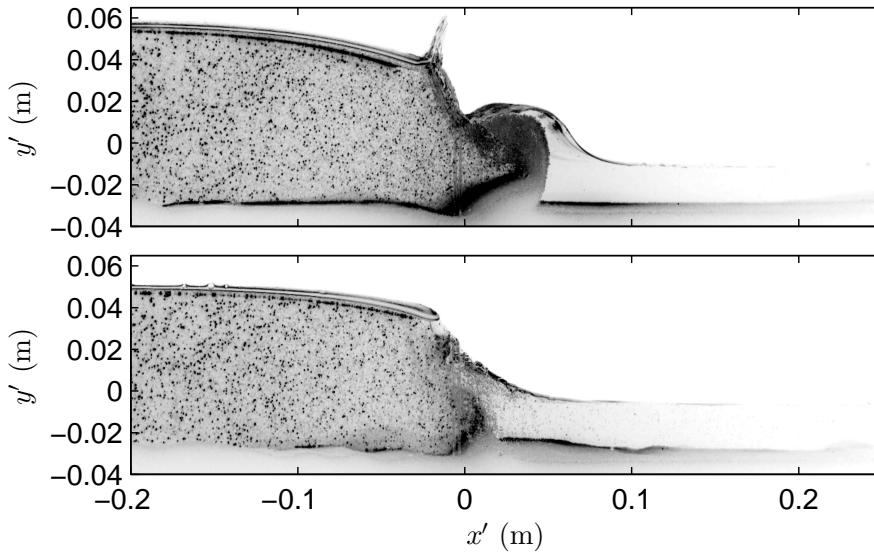


Figure 69: Comparison of photographic results for a tailwater over a sediment bed. (a) simple ($h_0 = 0.088\text{m}$ $h_1 = 0.017\text{m}$) and (b) wrapped-fabric release mechanisms ($h_0 = 0.076\text{m}$ $h_1 = 0.02\text{m}$). $t' = 0.15\text{s}$ where $t' = 0$ is the initiation of barrier removal.

Figure 71 and 72 show the velocity results of the two release mechanisms for a fully developed wave and for a wave during the early stages of collapse respectively. u_{\max} , v_{\max} and w_{\max} are the maxi-

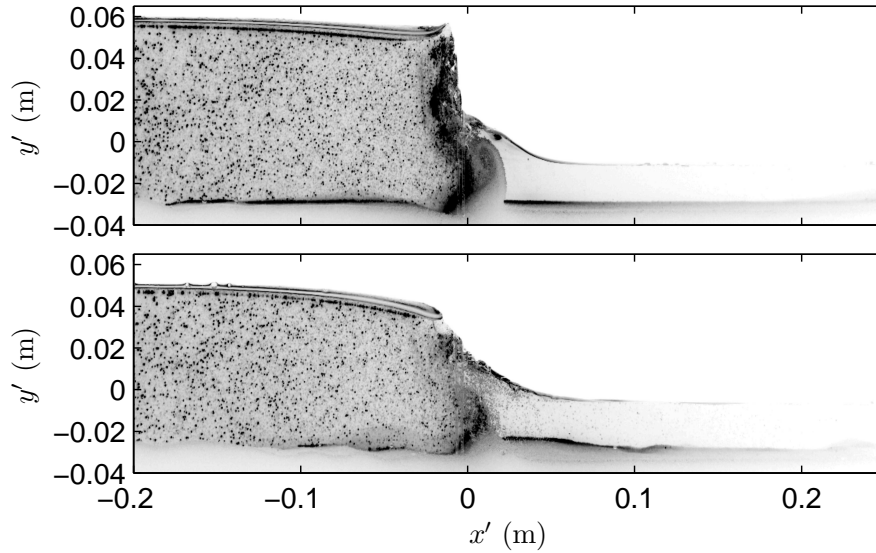


Figure 70: Comparison of photographic results for (a) simple and (b) wrapped-fabric release mechanisms, at $t = 0$ where $t = 0$ is the recession point of the lower edge from the fluid.

imum velocities in the x , y and z coordinate directions respectively, a water column at a given horizontal position. The subscripts “, simp” and “, fab” denote the simple release mechanism and the fabric release mechanism respectively. Figure 71 shows there to be almost no difference in the velocity profiles of the fully developed dam break waves released by the simple and wrapped-fabric release mechanism, for both the dry case and the tailwater case. Figure 72 compares the velocities of the two release mechanisms for the early time case ($t' = 0.02s$ after initiation of barrier motion for both methods). It shows that the wrapped fabric release mechanism reduced horizontal and vertical velocities, for both the dry and non-zero tailwater cases. This reduction exists only for a short time (until $t' = 0.04s$) before both release mechanisms profiles become closely comparable. Although a distinct difference can be seen for the early time case, the velocities involved are significantly smaller than the velocities exhibited by the fully developed dam break wave at later time. It is unknown how much the velocity differential and the shear related vorticity are each contributing to the difference in suspended sediment concentration.

Figure 71 also illustrates the results of the 3D-PIV. Both maximum and mean values of w were very small compared to the horizontal or even the vertical velocities. The velocities perpendicular to the laser plane (w) have been omitted from the following velocity plots for clarity. All results taken over a rigid bed produced negligibly small velocities perpendicular to the laser plane, confirming the assumption that the flow was predominantly two dimensional. The results produced by conducting the experiment over a sediment bed, resulted in large

velocities in all three component directions, where the resultant vector velocities pointed in random directions. These results were deemed not representative of the flow and as such will not be presented in this thesis.

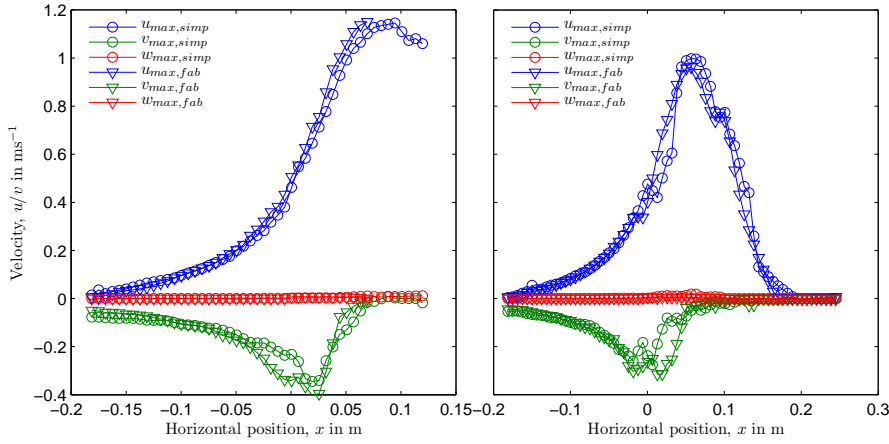


Figure 71: Maximum velocity plots (PIV) for (a) a dry bed ($t' = 0.15s$) and (b) a $h_{1,simp} = 0.0088m$, $h_{1,fab} = 0.007m$ tailwater case ($t' = 0.18s$), Fully developed wave

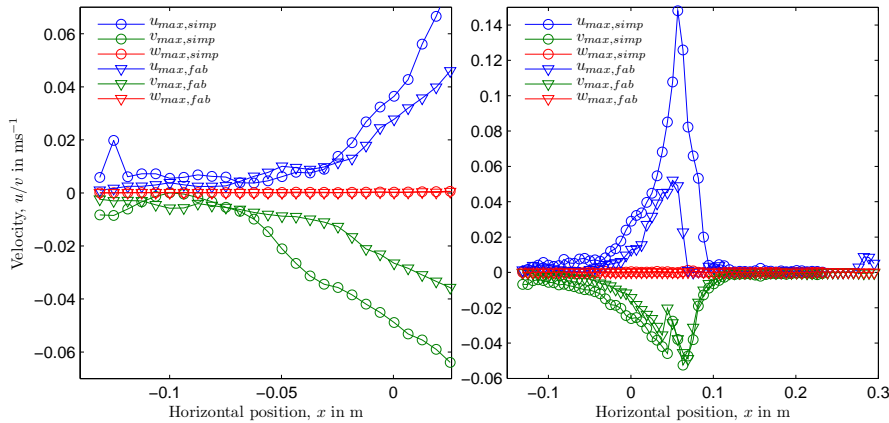


Figure 72: Maximum velocity plots (PIV) for (a) a dry bed and (b) a $h_{1,simp} = 0.0088m$, $h_{1,fab} = 0.007m$ tailwater case, Early time case, $t' = 0.02s$

4.2.2 Dry rigid bed

4.2.2.1 Initial conditions

The initial conditions for the zero tailwater or dry rigid bed test are illustrated by figure 64(a). The dam is located at $x = 0.2m$, where $x = 0$ is located at the cell edge between the simulation's zeroth and first column where the zeroth column is a ghost column for the imposition of boundary conditions. $x_f = 0$ at $x = 0.2$ for the analysis of the

front position. Backwater heights were chosen to exactly match the best experimental results for each case. As such, the backwater height equals 0.089m for the simple barrier system and the backwater height equals 0.094m for the fabric barrier system. The tailwater is set to zero for both cases. The dam is considered to have collapsed at $t = 0$. Initial velocities in both x and y directions are set to zero at $t = 0$ and the initial pressure in the fluid is hydrostatic, $p = \rho gh$ where h is the height of the water column above the point.

4.2.2.2 Results

Figure 73 shows the results for the horizontal front position evolution in time. The experimental results show a trend that closely matches the numerical results except for a small offset. If this is assumed to be related to the initial jetting phase, the results can be compared by subtracting the initial distance. The experimental results are in good agreement with both the no slip and free slip numerical results for over half of the studied time period. After this point the impact of resistance cannot be neglected and only the no slip show good agreement with the experimental results. This result confirms the statement made by Dressler [1954] that a no resistance assumption is valid for the initial stages of a dam break flow. Figure 74 shows the exper-

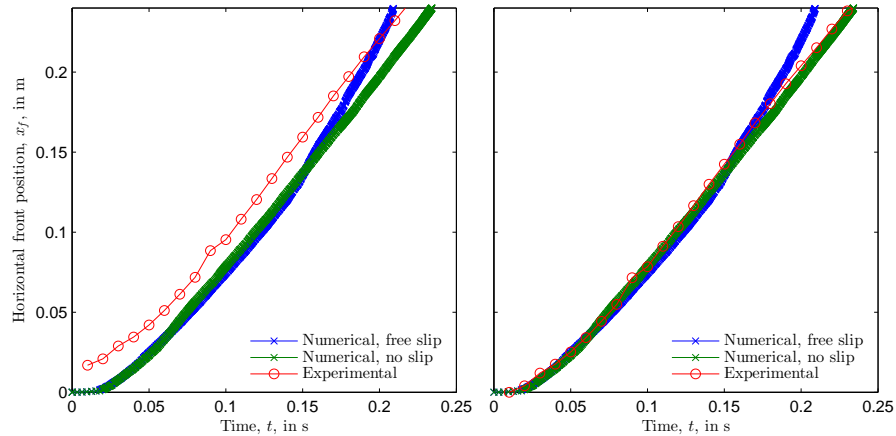


Figure 73: Horizontal front evolution for the dry bed case (simple release), with and without an applied offset.

imental and numerically produced surface profiles at various times. The numerical simulation differs the most from the experimental results for the early stages. This is because the observed experimental profile is the jet-like flow that occurs when a fluid is allowed to escape under a partially opened sluice gate. In an ideal release, the entire front face of the fluid block rotates, where a small advancement of the front raising fluid depth ahead of the dam and lowering it behind. In the jet like flow that occurs from the opening of a sluice gate, fluid is only allowed to move forward through the orifice area and

as such the resulting fluid heights ahead of the dam site are lower than they would be in the idealised case. Figure 74 shows reasonable agreement between the experimental and numerical results, with the numerical result tending to overpredict both the head angle and the flow depth behind the head region.

Figure 75 shows a comparison of the experimental surface profiles to those produced by numerical simulation using a free slip condition. The surface profiles more closely predict the experimental fluid depths, however they exhibit a greater number of surface irregularities.

Figure 76 compares the experimental surface profiles to the numerical results using a free slip condition, and the more aggressive flotsam and jetsam filling condition, where surrounded surface cells are filled in. This Figure shows that this condition had a large impact on the created surface profiles. The simulation produced smooth, regular surfaces, however it also significantly increased the head angle and flow depth ahead of the dam as compared to the experimental results. This showed that the application of fill-in measures to promote numerical stability can have significant impact on the numerically produced results. Thus, only the minimal fill in condition was employed for all other simulations.

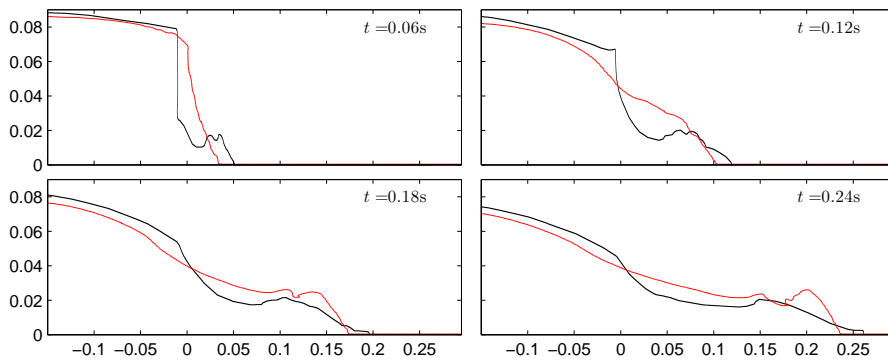


Figure 74: Surface Profile dry rigid bed. Numerical simulation with no slip conditions. Experimentally produced surface in black, numerically produced surface in red. Surfaces shown are the produced via a contour plot describing the 0.5 boundary

Figure 77 compares the horizontal and vertical velocities for the experimental and numerical (free slip) testing. It shows the maximum and mean velocities, where the mean is the average over the height for a given horizontal location x and the maximum is the maximum value over the height for a given horizontal location x . Numerical results tended to overpredict both horizontal and vertical velocities in the initial stages and have reasonable agreement between the maximum velocities for later times. Very good agreement can be seen for both maximum and mean values for velocities behind the dam site, however experimental mean velocities were consistently lower

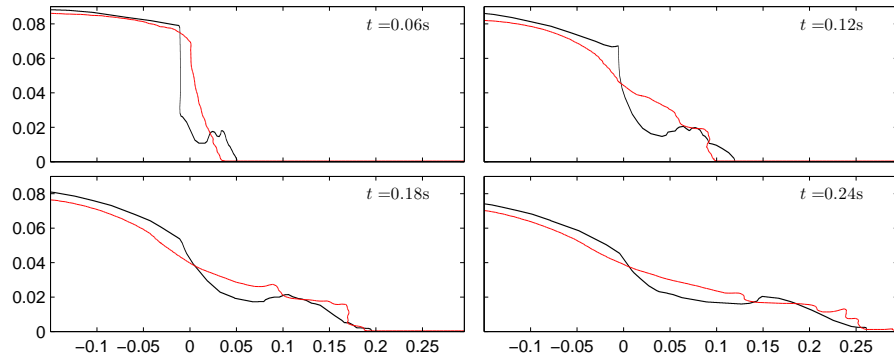


Figure 75: Surface Profile dry rigid bed. Numerical simulation with free slip conditions. Experimentally produced surface in black, numerically produced surface in red.

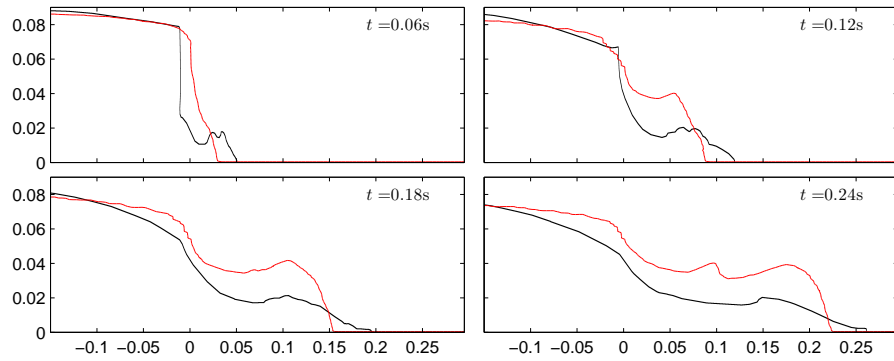


Figure 76: Surface Profile dry rigid bed. Numerical simulation with free slip conditions and the surface condition for the F/J void filler. Experimentally produced surface in black, numerically produced surface in red.

than the numerically predicted values. Figure 77 also shows a vertical velocity spike near the tip for the later times which was not observed in the experimental results. The comparison of the results of experiments run with the wrapped fabric release mechanism produced analogous results and as such have been omitted for clarity and simplicity. The wrapped-fabric release mechanism produced al-

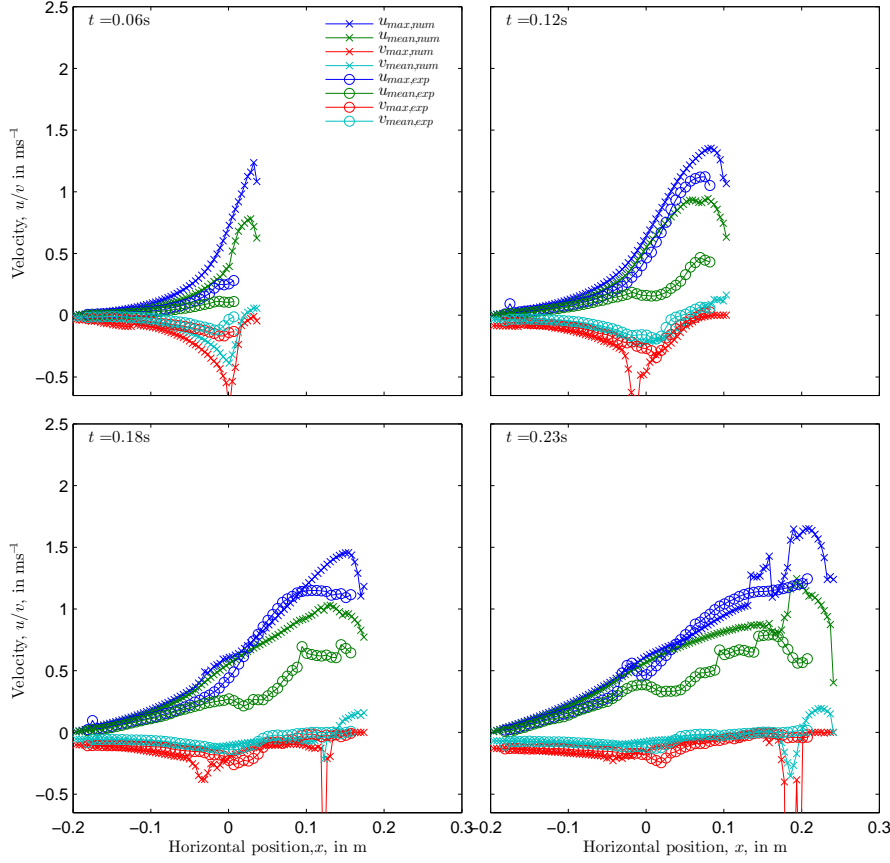


Figure 77: Mean and maximum horizontal and vertical velocities for the dry bed case (simple release) at $t' = 0.06s, 0.12s, 0.18s, 0.23s$.

most identical velocity profiles for all values of t , once adjusted for the release time differential.

4.2.3 Tailwater shallow

4.2.3.1 Initial Conditions

The initial conditions for the non-zero tailwater test and a rigid bed are illustrated by figure 64(b). The Simulation was set up identically to the dry rigid bed case, changing the backwater height to 0.094m and the tailwater height to 0.00875m for the simple release. The backwater height was set to 0.086m and the tailwater height to 0.007m for the fabric barrier system. The dam at $x = 0.2m$ with $x = 0$ located at the cell edge between the simulation's zeroth and first column where

the zeroth column is a ghost column for the imposition of boundary conditions. $x_f = 0$ at $x = 0.2$ for the analysis of the front position. The dam is considered to have collapsed at $t = 0$. Initial velocities in both x and y directions are set to zero at $t = 0$ and the pressure in the fluid is hydrostatic, $p = \rho gh$ where h is the height of the water column above the point.

4.2.3.2 Results

Figure 78 shows the results of the numerical simulation of a dam break wave over a tailwater compared to the experimental reconstruction of the same scenario. The experimental results showed that dam break over a tailwater creates a mushroom-like, protruding wave structure which forms a breaking wave. The numerical simulation will fail in cases of high curvature which occur during a breaking wave. The following presents the results up to the breaking of the wave where the simulations fails. Figure 78 shows good agreement between the experimental results and the numerical results produced with a free slip condition after the application of an offset.

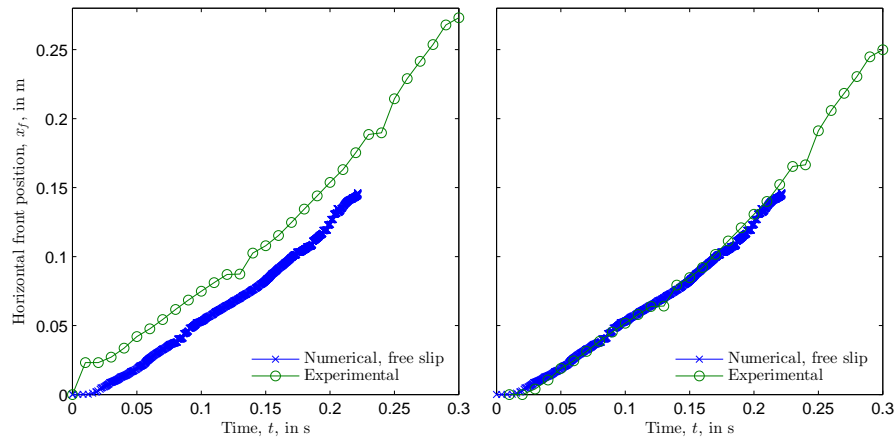


Figure 78: Horizontal front evolution for the non-zero tailwater case (simple release), with and without an applied offset.

Figure 79 compares the surface profiles of the experimental and numerical results (with a free slip condition) for various times. As with the dry bed, the initial stages show the largest difference in flow profile with the experiment producing a jet-like flow whilst the barrier is still in partial contact with the front face of the fluid. Close agreement can be seen between the results for later times, with the numerical simulation correctly predicting the formation of the protrusion and its growth into a breaking wave. As with the free slip results for the dry bed, some surface irregularities developed on the surface, particularly on the receding edge of the head.

Figure 80 shows the mean and maximum velocities for the tailwater at various times. Again, the numerical simulation overpredicted

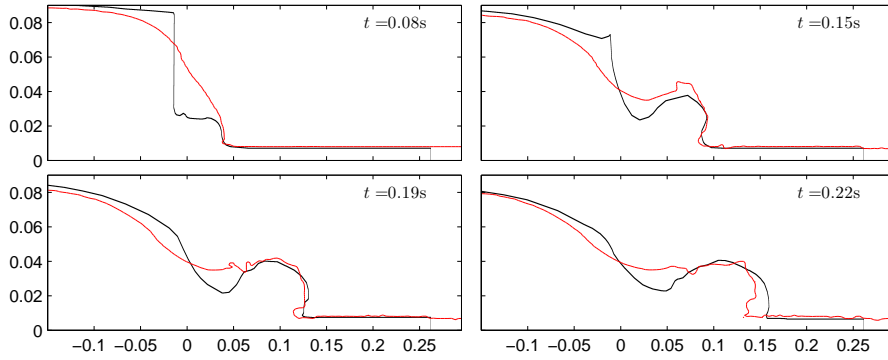


Figure 79: Surface Profile, tailwater, free slip. Numerical simulation with free slip conditions. Experimentally produced surface in black, numerically produced surface in red.

velocities during the early stages and roughly matched thereafter. Numerical results in general were approximately equal for maximum velocities but tended to over predict mean velocities beyond the dam site. Numerical results for the tailwater case showed a distinct reduction in horizontal velocity for a small region just behind the wave front. This was not observed in the experimental results. In addition the numerical results also produced some vertical velocity spikes which were also not observed in the experimental results.

4.2.4 Sediment

4.2.4.1 Initial conditions

The initial conditions for flow over the the non-zero tailwater test over a mobile sediment bed are illustrated by figure 64(c). The backwater height is equal to 0.088m for simple release and 0.076m for fabric and the tailwater height is equal to 0.017m for simple release and 0.02m for fabric release. The sediment was laid to a depth of 0.02m across the entire tank.

4.2.4.2 Results

The simulation of the dam break wave over a granular bed with a tailwater reached a high curvature failure condition very quickly. Figure 81 shows the front position evolution results of the experiment and the numerical simulation until its failure point. It can be seen that there was good agreement for front velocity until the simulation's failure. Due to this early failure, surface and velocity comparisons for fully developed waves cannot be made.

Results of dam break over a granular bed with a zero tailwater condition progressed significantly further before reaching a high curvature failure. Simulation of the zero tailwater case progressed long enough to excavate a trough in bed height around the dam location

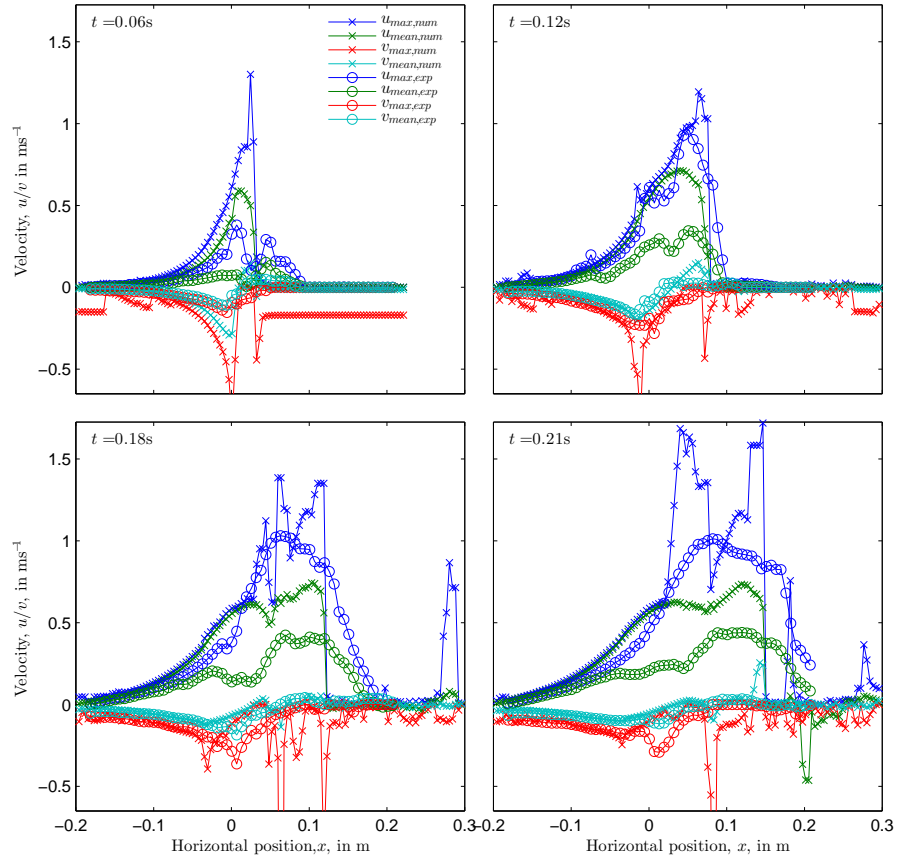


Figure 80: Mean and maximum horizontal and vertical velocities for the non-zero tailwater case (simple release) at $t' = 0.06s, 0.12s, 0.18s, 0.21s$.

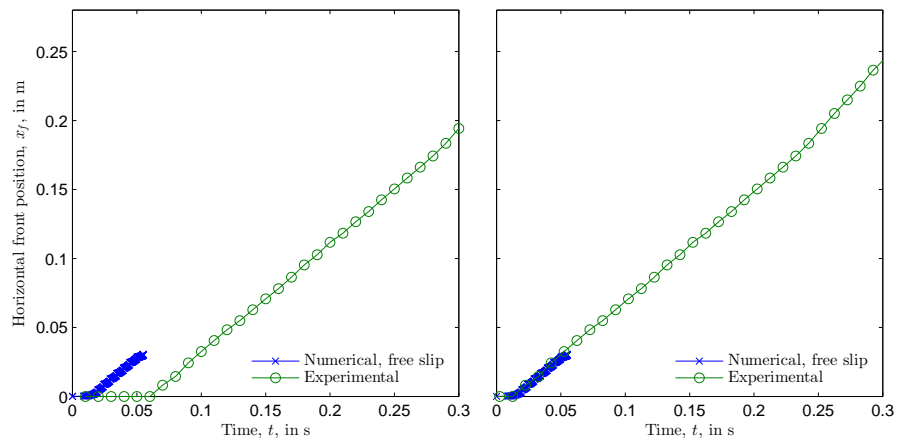


Figure 81: Horizontal front evolution for the non-zero tailwater over a granular bed case (simple release), with and without an applied offset.

($x = 0$). The results of this simulation are presented in figure 82. Figure 82 shows that when the heights within the trough are properly determined to be and redistributed as a trough, the bed height profile produced shows good agreement with the shape and location of bed surface profile produced in the experiment (see figures 69 and 70). The numerical simulation, however, suspended a significantly greater quantity of sediment which can be seen by the bed height profile which is much lower in the numerical results.

Figure 83 and figure 84 show the results of a simulation with the same initial conditions, where the value of P_{bool} has been manipulated to demonstrate the consequence of a misinterpretation of a trough as a peak. Figure 83 is the result before the first redistribution and figure 84 is the result after the first redistribution. The result is the raising of the bed beyond the original bed height in the region behind the dam, which was not observed in the experimental results.

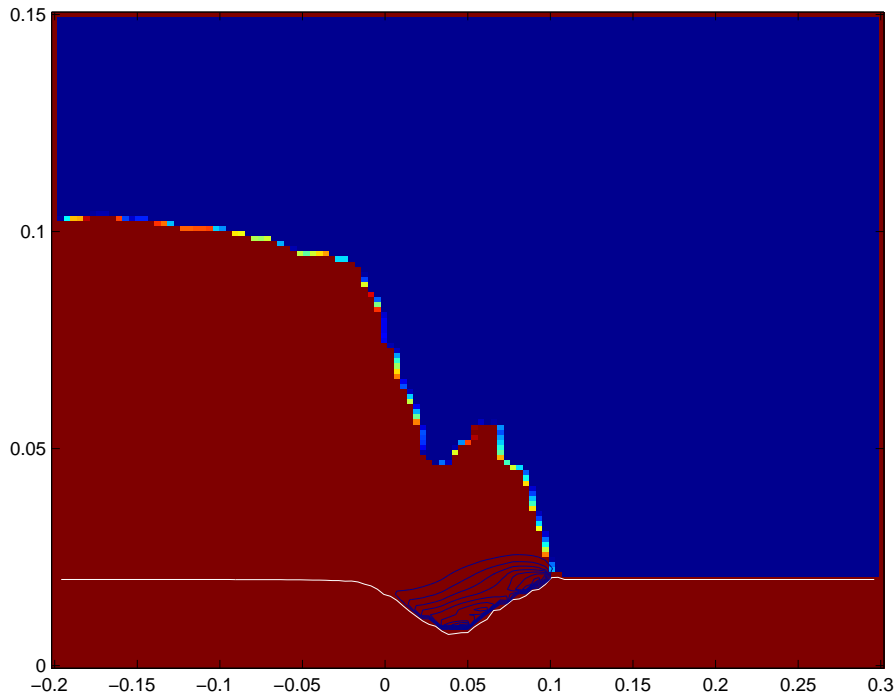


Figure 82: Sediment over dry bed at $t=0.14s$, The colour red represents volume fractions equal to 1, the colour blue represents volume fractions equal to zero. The White line represents the bed height and the contour plot shows the concentration of suspended sediment. Volume fractions equal to 1 above the white bed height line represent cells filled entirely with fluid. Volume fractions equal to 1 below the white bed height line represent cells filled with sediment at the packing density where voids are filled with water.

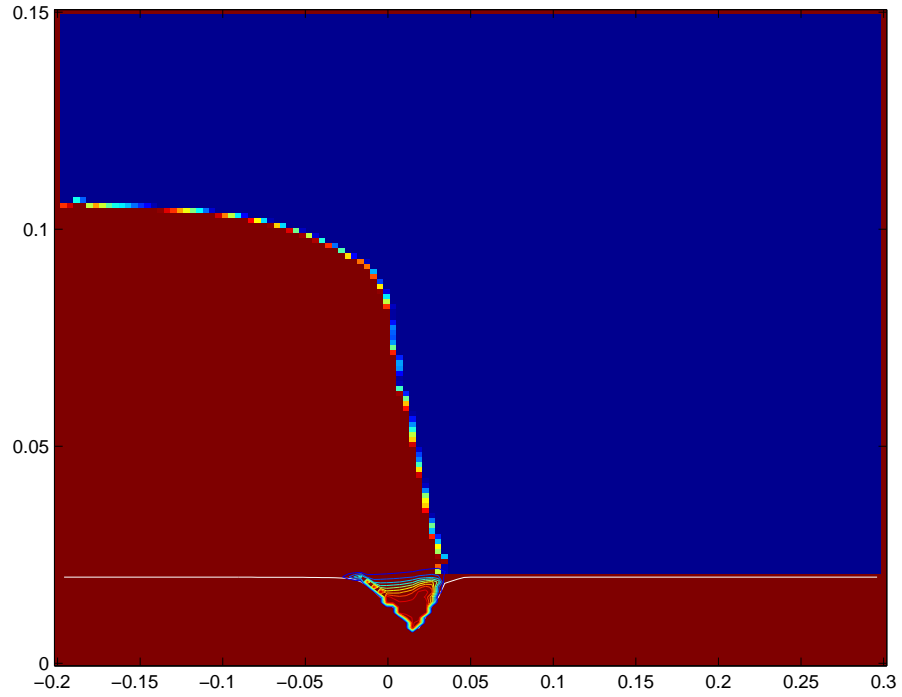


Figure 83: Sediment over dry bed, surface profile, bed profile and concentration. Separate simulation to force misinterpretation of trough. Before. $t' = 0.0861s$

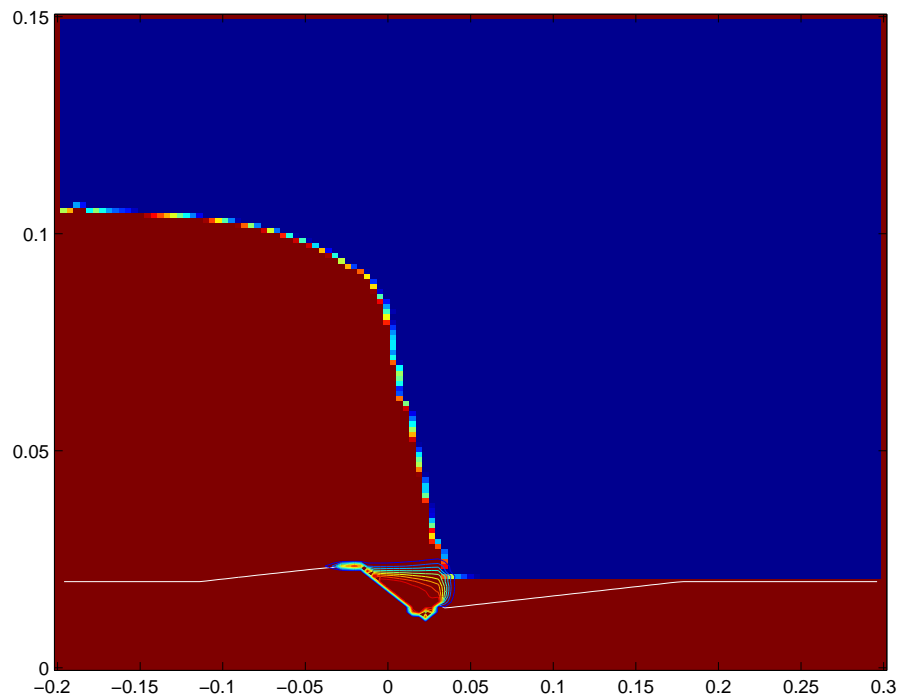


Figure 84: Sediment over dry bed, surface profile, bed profile and concentration. Separate simulation to force misinterpretation of trough. After. $t' = 0.0861 + 4 \times 10^{-5}s$

SUMMARY AND CONCLUSIONS

5.1 SUMMARY OF THESIS

The first academic study, an analytical solution, of dam break flow was published in 1892 by Ritter. Since then, the problem was adapted to include bed resistance [Dressler, 1952, 1954; Whitham, 1955; Hogg and Pritchard, 2004], the initial stages [Stoker, 1957; Korobkin and Yilmaz, 2009], bed discontinuities [Bernetti et al., 2008], and now mobile beds. Numerical simulation, in particular, of dam break over mobile beds, has become an area of concentrated interest [Capart and Young, 1998; Fraccarollo and Capart, 2002; Cao et al., 2004; Wu and Wang, 2007; Zech et al., 2008; Kelly and Dodd, 2009; Xia et al., 2010; El Kadi Abderrezzak and Paquier, 2010; Zhang and Duan, 2011; Soares-Frazão et al., 2012; Zhang et al., 2014; Marsooli and Wu, 2014; Razavitoosi et al., 2014; Kesserwani et al., 2014], with a large number of somewhat similar models, producing mostly comparable results Soares-Frazão et al. [2012]. The present work has focussed on the development of new numerical methods for the study of fluid dynamics and sediment transport. This included a new sediment bed to fluid bulk transition method and a novel avalanching model for the maintenance of a bed slope limit, for the coupling of a sediment transport model with the solution of the Navier-Stokes equations for an unsteady open channel flow in two dimensions. The motivation for this work was for its application to the study of dam break flow and its potential further applications to coastal morphodynamics and beach face evolution modelling. The free surface evolution, velocity profiles and bed evolution morphodynamics have been examined for the idealised dam break of a finite volume over flat, rigid beds and erodible beds.

Experimental study of dam break began in 1917 with [Schoklitsch, 1917a], however the amount of data presented in this original paper was somewhat limited. A complete data set was first presented by Eguiazaroff in 1935 and an equivalent data set for a variety of bed resistances was presented by Dressler [1954]. Concurrently with mathematical study, experimental study then focussed on the initial stages Stansby et al. [1998] and recently on dam break over movable beds [Capart and Young, 1998; Leal et al., 2001; Fraccarollo and Capart, 2002; Spinewine, 2005; Spinewine and Zech, 2007; Zech et al., 2008; Soares-Frazão et al., 2012]. Primarily, these studies have used probes and or photography to determine the evolution of the free surface profile as well as find the final state of the bed. Only recently have

attempts been made to better track the velocity field of the flow and the sediment particles. There has been very little discussion within the literature of the potential impacts of disturbance caused by the rapid, shear inducing withdrawal of the reservoir retaining barrier. The present work was primarily focussed on the production and implementation of a numerical model, however in addition to this a run of experiments was conducted to provide experimental validation of the model and to investigate a new disturbance reducing barrier removal technique, based on apparatus used by Dalziel [1993] in the study of Rayleigh Taylor Instability. These were conducted using a 3D Particle Image Velocimetry (PIV) technique, which had not been attempted until this point for this type of flow.

5.2 THESIS CONCLUSIONS

The numerical model was presented with the results of its application to simple test cases for validation. The Navier-Stokes solver was tested on a lid driven cavity problem, a square, initially still domain, with a specified tangential velocity at the upper boundary or lid. The results converged, closely matching the solutions published by Ghia et al. [1982], for all presented methods on coarse grids, however performance reduced significantly for the simpler Gauss-Seidel and Successive over relaxation methods. The free surface reconstruction was tested through the application of various test objects and flow fields. The results had close agreement to the results presented by Rudman [1997], on which the method is based. The sediment transport model was tested by applying various initial conditions so as to observe the settling of sediment through a domain, the generation of sediment concentration profiles for equilibrium conditions and the collapse of sediment bed profiles for the maintenance of a critical angle of repose.

Two release mechanisms were tested for the experimental reconstruction of the dam break test case. The first was the simple vertical removal of a flat barrier. The second used a tethered flexible fabric, wrapped around the lower edge of the barrier, to indirectly remove the barrier by imposing the vertical impetus to the fabric rather than the barrier. The wrapped fabric mechanism, inspired by a similar method used in the study of Rayleigh-Taylor Instability by Dalziel [1993], had various impacts on the flow. The wrapped fabric mechanism consistently required a greater amount of time to fully remove the barrier. This caused some difficulty in direct comparison of the results of the two methods. A time offset was determined to match the front positions. In most cases the offset was determined by redefining the time t to be zero at the moment the barrier's lower edge is no longer in contact with the liquid, however in some cases this produced an over correction. The wrapped-fabric method consistently produced flow depths beyond the dam site, greater than the depths

produced by the simple release, with a steeper angle between the head of the flow and the bed. This is thought to be due to the more gradual release of the water behind the barrier. The wrapped-fabric method also caused a reduction in the amount of sediment that is suspended into the flow. This may be due to the reduction in the shear related vortical disturbance, however it may also be due to a reduction of fluid velocity at the initiation of flow caused by the more gradual release.

The numerical simulation was shown to have good agreement with experimental results for the dam break over a dry bed. Simulations with a non slip condition produced the best agreement for the horizontal front location and velocity profiles. The reconstructed free surface had less agreement with the experimental results. This is thought to be due to the overly simplified free surface conditions and the lack of surface tension. The surface profile was strongly affected by the flotsam and jetsam in-fill conditions and as such the extent of the application of these conditions was limited to the minimum required to provide numerical stability.

The simulation performed less well over a wet bed, where the fluid formed a breaking wave. The complex geometry and tight curvature involved in a breaking wave surface caused the simulation to fail. The results up to failure are presented and showed reasonable agreement until a short time before the model failure. Within the shortened period, free slip conditions provide good agreement with the numerical results.

The numerical simulation of a dam break over a granular bed (with zero and non-zero tailwater cases) performed poorly, failing to produce a solution for the flow properties after a short period of time. This is because of the tight curvatures of the breaking waves that form under these conditions. Initial comparisons are made between the numerical model and experimental results for the granular bed case. Some agreement can be seen for the location of erosion and the shape of the bed forms created, however the total amount of erosion and sediment suspension is overpredicted by the numerical simulations. Comparisons of final bed profiles could not be made due to the early failure of the simulations.

5.3 RECOMMENDATIONS FOR FUTURE WORK

5.3.1 *Numerical*

The numerical study gleaned various methods however it has a limited scope for potential application in its current form. The simulation still relies on free surface boundary conditions which are approximate and limiting for applications. The current free surface boundary conditions require the curvature of the free surface to not be tight. In

the case of dam break over a dry bed this requirement holds, however dam break over a tailwater has been shown both numerically and experimentally to produce a breaking wave formation. The tight curvature of the breaking wave and the complex geometry that develops when a breaking wave attempts to recombine with the tailwater caused the simulation to consistently fail at or near the point of wave breaking. Some small improvements were gained by applying an adapted form of the flotsam and jetsam remover to fill in and interpolate the small voids created by the complex interactions, however these were found to be insufficient and the simulations continued to fail.

The largest failing of the present model is in its application of free surface boundary conditions. The present model uses a adaptation of the free surface boundary conditions used in the GENSMAC method [Tome and McKee, 1994], however this method does not attempt to calculate the surface angle or curvature in the cell and apply these to more accurately determine the surface pressure and velocity conditions to balance the normal and tangential stresses at the surface. The present model does have the capability to calculate a surface angle for the cell from a stencil of the volume fractions in the surrounding cells, however to apply surface tension at the boundary, a separate method would need to be applied to calculate the surface curvature from the volume fractions. Three methods for estimating curvature from Volume fractions are presented by Cummins et al. [2005], including The CV method [Williams, 2000], the HF method [Sussman, 2003] and the RDF method [Cummins et al., 2005]. Once the curvature is calculated, the capillary pressure difference or the pressure difference due to surface tension across the interface can be calculated by applying the Young-Laplace normal stress balance equation

$$\Delta p_c = \gamma \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \quad (109)$$

where Δp_c (sometimes given solely as p_c) is the pressure difference across the interface due to surface tension, or the capillary pressure. γ is the surface tension (SI unit, N m^{-1}), and R_1 and R_2 are the radii of curvature in two directions, for a three dimensional surface ($R_2 = 0$ for a two dimensional flow surface). The simulation failed to converge to a solution for dam break over a tailwater as the curvature became too great for the simplified boundary conditions. Calculating surface location and curvature would allow for surfaces with greater curvature to still be considered, with accurate imposition of pressure and velocity boundary conditions at the surface. Areas of high curvature are still potentially susceptible to issues where the velocity applied externally to account for tangential stress, overwrites another closely located cell's external velocity. Further study would be required to find a solution to this issue. In addition, it is thought that the differ-

ences between numerical and experimental surface profile results at the front may have been caused by not accounting for surface tension.

The inclusion of surface tension would require some re-evaluation of contact angle determination. The present model does not directly impose a contact angle, but rather allows the volume fraction free surface reconstruction method to reconstruct the interface in the tip cell, whilst considering the bed to be full cells. This produced contact angles just under 90° . If surface tension were to be considered at the tip, the contact angle would need to be calculated. The contact angle under static conditions is given by Young's equation [De Gennes, 1985]

$$\cos(\theta_c) = \frac{\gamma_{SG} - \gamma_{SL}}{\gamma_{LG}} \quad (110)$$

where γ_{SG} is the solid-gas interfacial energy, γ_{SL} is the solid liquid interfacial energy, γ_{LG} is the liquid-gas interfacial energy or surface tension and θ_c is the contact angle. This cannot be directly applied to a simulation, as the bed surfaces have a roughness that needs to be considered, and the fluid will not be at rest. The contact angle of an advancing fluid surface will be larger than the at-rest contact angle. Rather than trying to determine a surfaces interfacial energy or wetting properties, it is recommended that the contact angle be directly measured and imposed in the simulation directly. This would be enacted by using the measured contact angle in place of the surface reconstruction angle calculation for the leading tip cell.

The experimental results showed that the dam break wave (over a dry bed) front tip always had a contact angle of approximately 90° , with the height at the tip often being very small. This validates the present model's use of a contact angle approaching 90° for the dam break case.

It must be noted that the inclusion of sediment concentration within a flow could have a large impact on surface tension and the other interfacial energies. The present model does not account for this and further research is required in this area to fully couple the models.

One avenue for further development of the numerical model would be to remove the constant density assumption used in the solution of the Navier-Stokes equations, and to couple this with the sediment calculations, so that there may be an investigation on how the suspended sediment affects the flow. The model in its current form uses the flow properties to impact the motion of the sediment, however the impact of the suspended sediment back on the flow is not well characterised. Finally the inclusion of a turbulence model would also allow for the more accurate analysis of the flow behaviour and a sediment transport calculation, based on fewer approximations that do not hold for realistic and unsteady scenarios. Further studies might look to apply some of the geometric methods developed in this thesis to more developed and robust commercial software, where turbulence closure

and density variation has already been included, and where well-mannered numerical solution methods for these cases have already been applied.

5.3.2 *Experimental*

It must be concluded that the experimental methods used in an attempt to validate this work were found to be not well suited to the experiment to which they were applied. The 3D-PIV image processing methods used to calculate the velocities in the fluid, required the Pliolite particle reflections to be perfectly exposed, with little to no other exposure existing. This was found to be impossible to reliably reproduce as the interaction of the surface of the flow with the laser sheet would cause unpredictable reflections. This resulted in identical experimental runs producing significantly different exposure conditions for the camera. This would manifest as an overall over or under exposure, or as artefacts, glints of light from the various reflections, that would interfere with the correct exposure of the image.

The experimental failure rate was reduced by angling the laser light source through the fluid such that it contacted the free surface last, however this was only successful for a small proportion of the experimental runs. Image masking and various image processing techniques were employed to improve the images and increase the chance that the PIV algorithms would converge to a solution, however visual artefacts significantly reduced the success rate of the method. Approximately 100 experimental runs were conducted for a variety of backwater and tailwater heights and for rigid and mobile bed conditions. Only a very small minority of the total number of experiments conducted, produced velocity data. Approximately 15 % of experimental runs produced a result that would allow the 3D-PIV algorithm to converge to a solution and less than 70 % of converged solutions produced realistic velocity profiles. The successfully converged velocity results were presented in section 4, however the sample size is extremely small. Thus the results can only be taken as indicative and needing further experimental validation.

PIV was found to be completely unsuccessful over a sediment bed. The sediment bed provided another irregular surface with which the laser could interact, producing further visual artefacts. Suspended sediment proved problematic as it would impede the camera's view of the Pliolite tracking particles and block the light of the laser, casting areas of the fluid into darkness. No sediment bed runs produced sensible velocity data.

The Dalziel fabric or flexible sheet barrier removal system was found to produce three main problems. For the distance the barrier was required to move to fully leave the tank, the fabric system was required to move twice as far. This caused a reduced and more inconsis-

tent barrier removal rate when using the fabric system. The increase in time required to remove the barrier caused an increase in volume that was allowed to jet out from underneath the barrier as it was being removed, producing a slightly different flow type to that of an idealised dam break flow. Future comparisons of these methods should attempt to equalize the removal rates by reducing the removal rate of the simple vertical removal. Secondly, the wrapped fabric method can easily trap liquid between the fabric and the barrier, which can spill out causing an unpredictable disturbance to the flow. Finally the wrapped barrier system caused the system to no longer be perfectly watertight. This is unacceptable for the reconstruction of dam break over a dry bed, where no liquid can be allowed to pass the barrier until the fluid is released.

Future experimental studies of dam break interaction with sediment beds would need to reconcile the issues of quantifying suspended sediment motion with capturing flow velocities. It was found that 3D PIV is unsuitable both for reliably capturing flows with a rapidly changing free surface and for capturing flows which contain a large quantity of suspended sediment. These issues were found to be significantly worse for three dimensional PIV as the method is very sensitive to experimental errors and artefacts. In addition, the angling of the cameras required for PIV also increases the tendency toward light reflection artefacts experienced by at least one of the two cameras.

One option for the quantification of sediment transport would be to conduct a series of experiments in an open ended flume. The experiment would use a movable gate insert and matching back wall insert, to create a specific backwater length at various distances from the open end of the flume. The open end will allow the dam break flow (and its suspended sediment contents) to be collected, dried and weighed. This would provide an instantaneous total suspended sediment mass for the flow for the specific distance from the dam. This would enable data collection for suspended sediment load simultaneously with data collection on bed morphology via photography through a clear side wall.

Part I

APPENDIX

SUPPLEMENTAL INFORMATION

A.1 ADDITIONAL FREE SURFACE METHODS

A system of boolean flags were used to monitor each cell's status, showing whether the cell was either boundary cell or a control volume cell, whether or not the cell contained fluid, whether the cell contained a free surface and whether or not the cell was part of the bed.

A.1.1 *Marker and Cell*

The simplest method of keeping track of the fluid is to use a marker and cell method. This uses virtual particles, dispersed throughout the fluid to follow the flow. The method does not, however, track the free surface. An approximation of the free surface location can be made by tracking the last line of virtual particles. Virtual particle locations are progressed by taking each particle coordinate location, determining which cell it is located within and so which velocities are acting upon it. At this point the new particle locations are as follows: [Amsden and Harlow \[1970\]](#).

$$\frac{dx}{dt} = u, \quad (111)$$

$$\frac{dy}{dt} = v \quad (112)$$

$$x^{n+1} = x_p^n + u\delta t \quad (113)$$

$$y^{n+1} = y_p^n + v\delta t \quad (114)$$

The process of locating the particles within the grid is computationally costly, and the method doesn't accurately track the free surface, making the method not ideal, however in the case that you wish to be able to track the movement of virtual particles throughout a moving fluid, a useful technique for the visualisation of flows, the method is quite useful.

A.1.2 *SLIC*

[SLIC](#) is the simplest volume-fraction tracking method, developed by [Noh and Woodward \[1976\]](#) that maintains a field of volume fractions, c , for the entire grid representing the fraction that each cell is filled

with a particular fluid. These methods, and this method in particular, are good at keeping track of multiple fluids interacting. SLIC defines an index with two boolean values (I_L, I_R) that refer to whether the fluid exists in the cells either side of the cell in question. When dealing with a single fluid system, an index of (1,1) is a horizontal interface, an index of either (0,1) or (1,0) is a vertical interface and an interface of (0,0) is a fluid sandwiched between interfaces. For a two fluid system, when both fluids have the same index then the interface is set to be horizontal. If one index is different to the other and either one is equal to (0,1) or (1,0) then the interface is deemed to be vertical. If the indices are different but neither is equal to (0,1) or (1,0) (i.e. one is equal to (1,1) and the other is (0,0)), then the fluid with index (1,1) is sandwiched between. A three fluid system has two additional scenarios, that where a fluid is sandwiched between two different fluids or when three fluids meet with a 'T' shaped interface.

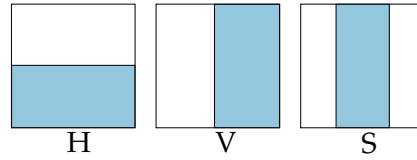


Figure 85: SLIC one or two fluid cases

A.1.3 Chorin's Method (FLAP)

Chorin's FLAP method [Chorin, 1980] is an adaptation of the SLIC method for one or two fluids that allows for small rectangles of fluid to exist at cell corners, creating a 'stair stepped' interface. These exist when the neighbouring cells on two adjacent sides contain fluid and the other two sides do not. The rectangle geometry can be determined by using the condition $ab = c_{i,j}$ and $\frac{b}{a} = \frac{c_{i-1,j}}{c_{i,j-1}}$ where a and b are equal to the proportion of the cell edge in contact with fluid for the two sides adjacent to cells containing fluid. For example, in the case of a cell where a rectangle of fluid will be placed in the bottom left hand corner of the cell, the equations for the filled edge proportions are

$$s_b = \sqrt{\frac{c_{i,j}c_{i,j-1}}{c_{i-1,j}}} \quad (115)$$

$$s_l = \frac{c_{i,j}}{s_b} \quad (116)$$

where s_b is the bottom edge filled proportion, or the proportion of the bottom cell edge in contact with fluid in the cell and s_l is the left edge filled proportion. Similarly if we have the case where a rectangle of fluid is omitted from a cell corner we can calculate the geometry of the rectangle and the associated filled proportions by using the

conditions: $ab = (1 - c_{i,j})$ and $\frac{b}{a} = \frac{(1 - c_{i-1,j})}{(1 - c_{i,j-1})}$ where a and b are now equal to the proportion of the cell edge not in contact with fluid in the cell. For example, in the case of a cell with a rectangle omitted from the top right corner, the equations for the filled proportions are:

$$s_t = \sqrt{\frac{(1 - c_{i,j})(1 - c_{i,j-1})}{(1 - c_{i-1,j})}} \quad (117)$$

$$s_r = 1 - \frac{(1 - c_{i,j})}{(1 - s_t)} \quad (118)$$

where s_t is the top edge filled proportion and s_r is the left edge filled proportion. Using this we can formulate eight corner cases, four where the fluid is in a block in each of the four corners and four where an absence of fluid is located within a rectangle in each of the four corners. In addition to these we also explicitly define the case of a fluid finger, or the fluid enclosing a finger of open space. Note that in a two fluid system, this absence can also be the second fluid. All of these cases are illustrated in figure 86.

A.2 INITIAL CONDITIONS-CIRCLE

The calculation of accurate initial conditions for a circle or Zalesak slotted disk for the testing of a free surface solver requires the following procedure. For any given cell intersected by the circle edge, the location of the two intersection points must be found. Using this information it is possible to calculate the area of the cell bisected by a straight line going through these two points, A_c . It must be noted that this area is approximately correct however, underestimates the actual area by the area between the straight line and the arc. To calculate the actual area, the area of the arc sector, A_{arc} , bounded at each intersection point with lines radiating from the circle centre must be calculated. The area of the triangle, A_{tri} , made up of the two radii and the straight line between the edge intersection points must also be calculated. The additional area that is to be added to the original approximation is equal to the area of the arc minus the area of the triangle.

The following equations are those required to define the volume fractions of cells where the interface is defined by a circle of radius r , for cells in the upper left quadrant of the circle. Assuming the circle is centred on a cell corner, these values can be mirrored to form the rest of the circle. Note that the interface scenarios required for the

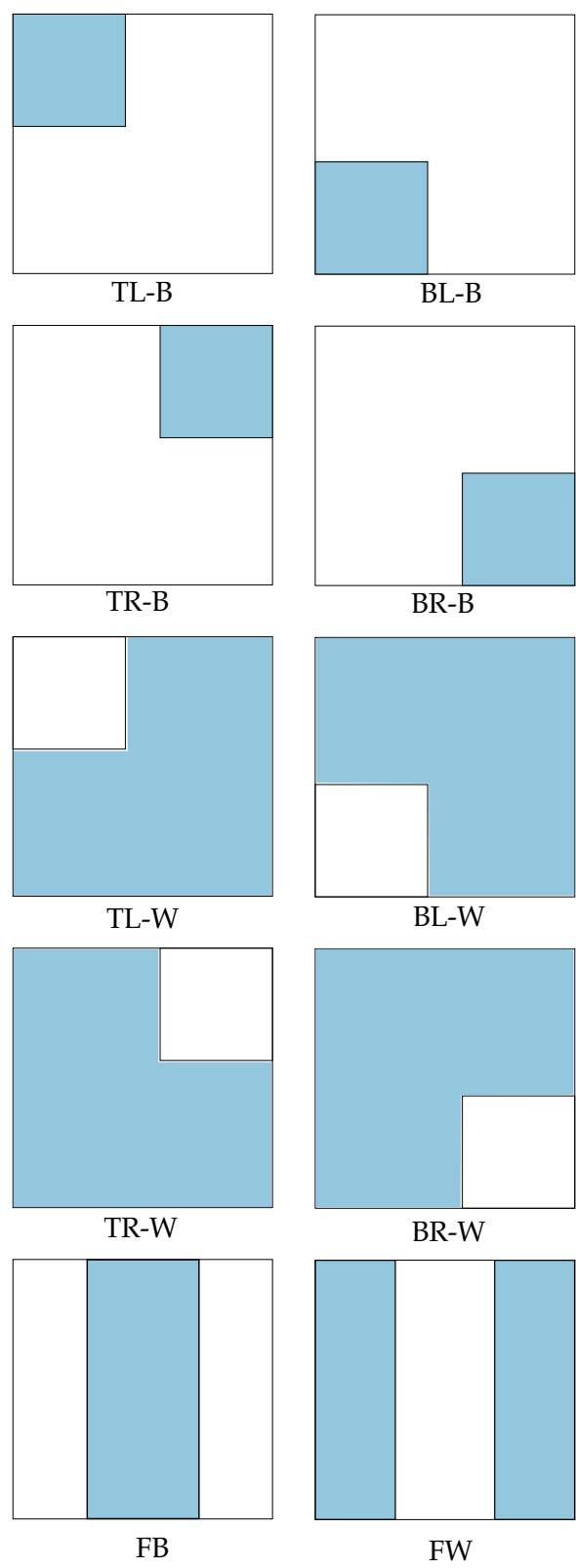


Figure 86: Chorin's corner cases

upper left quadrant are equivalent to the Orientation A, Case I-IV cells given by Young's method.

$$\begin{aligned}
b_{AI} &= \frac{sr\delta y}{\sin\left(\arctan\left(\frac{sr\delta y}{sb\delta x}\right)\right)} \\
A_{lin,AI} &= \frac{sr\delta ysb\delta x}{2} \\
b_{AII} &= \frac{(sr-sl)\delta y}{\sin\left(\arctan\left(\frac{(sr-sl)\delta y}{sb\delta x}\right)\right)} \\
A_{lin,AII} &= sl\delta x\delta y + \frac{(sr-sl)\delta y\delta x}{2} \\
b_{AIII} &= \frac{\delta y}{\sin\left(\arctan\left(\frac{\delta y}{(sb-st)\delta x}\right)\right)} \\
A_{lin,AIII} &= st\delta x\delta y + \frac{(sb-st)\delta y\delta x}{2} \\
b_{AIV} &= \frac{(1-sl)\delta y}{\sin\left(\arctan\left(\frac{(1-sl)\delta y}{(1-st)\delta x}\right)\right)} \\
A_{lin,AIV} &= \left(1 - \frac{(1-sl)(1-st)}{2}\right)\delta x\delta y \\
A_{arc} &= \pi r^2 \left[\frac{2 \arcsin\left(\frac{0.5b}{r}\right)}{2\pi} \right] \\
A_{tri} &= 0.5b^2 \sqrt{\frac{r^2}{b^2} - \frac{1}{4}} \\
c_{i,j} &= \frac{A_{lin} + (A_{arc} - A_{tri})}{\delta x\delta y}
\end{aligned}$$

A.3 INITIAL CONDITIONS-ROTATED SQUARE

Define the corners of the unrotated square with respect to an origin at the square centre. Rotate the corner points by using the following equations.

$$x_n[a] = x[a] \cos(\alpha) - y[a] \sin(\alpha) + x_{centre} \quad (119)$$

$$y_n[a] = y[a] \cos(\alpha) + x[a] \sin(\alpha) + y_{centre} \quad (120)$$

Next we calculate the area in the cell where the points exist. This is accomplished by splitting the area in two at the point. The exact equation for these areas is dependant on which of the cell edges the interfaces cross. For example at the left hand point, if the upper inter-

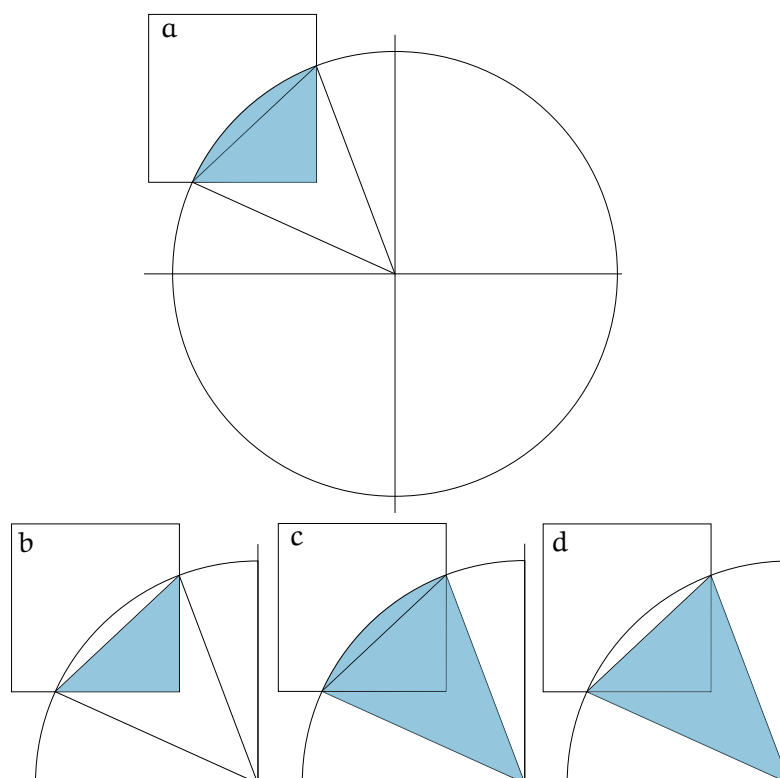


Figure 87: Areas to be calculated to calculate a circular initial condition. Figure a) Shaded area represents A_c , Figure b) shaded area represents A_{lin} , c) shaded area represents A_{arc} and Figure d) shaded area represents A_{tri}

face crosses the upper cell edge, and the lower interface crosses the right hand cell edge the equation for the upper area is

$$a_1 = (0.5((1 - s_x) - s_t)\delta x(1 - s_y)\delta y) + s_t\delta x(1 - s_y)\delta y \quad (121)$$

$$a_2 = 0.5(((1 - s_x)\delta x)((1 - s_x)\delta x)) \tan(\beta) \quad (122)$$

$$c_{i,j} = \frac{a_1 + a_2}{\delta x \delta y} \quad (123)$$

Once the volume fractions at the points are set, the algorithm follows an interface one cell width at a time calculating the edge intersections and the areas of the geometries that they create. This is done in a fashion similar to that of the Young's method. The young's cases I, II, III IV are used to describe the four different intersection types that can exist.

A.4 USER INTERFACE

The model as described above was written into a program in C++, for more information on the specific algorithms created to do so see Appendix B. In addition to this, a user interface was written using Windows Forms, for the simple usage of the program. The procedure for using the program is to double click on the Forms application executable located within the Release folder. This will launch the initial conditions set up, see figure 88. Once the fields have been filled out with the desired parameters, the user will click Go, launching the second panel, see figure 89, which creates the argument pass through file which is read by the simulation. Once this has been filled out (note that values already filled on the previous panel will appear in their fields on the second panel) the user will once again press go. The user is now ready to double click the simulation executable which is located in its own Release folder. The simulation will run, and the results of the simulation will be automatically exported to the location given by the user in the forms application. That location will also include an error log file for any errors thrown by the simulation.

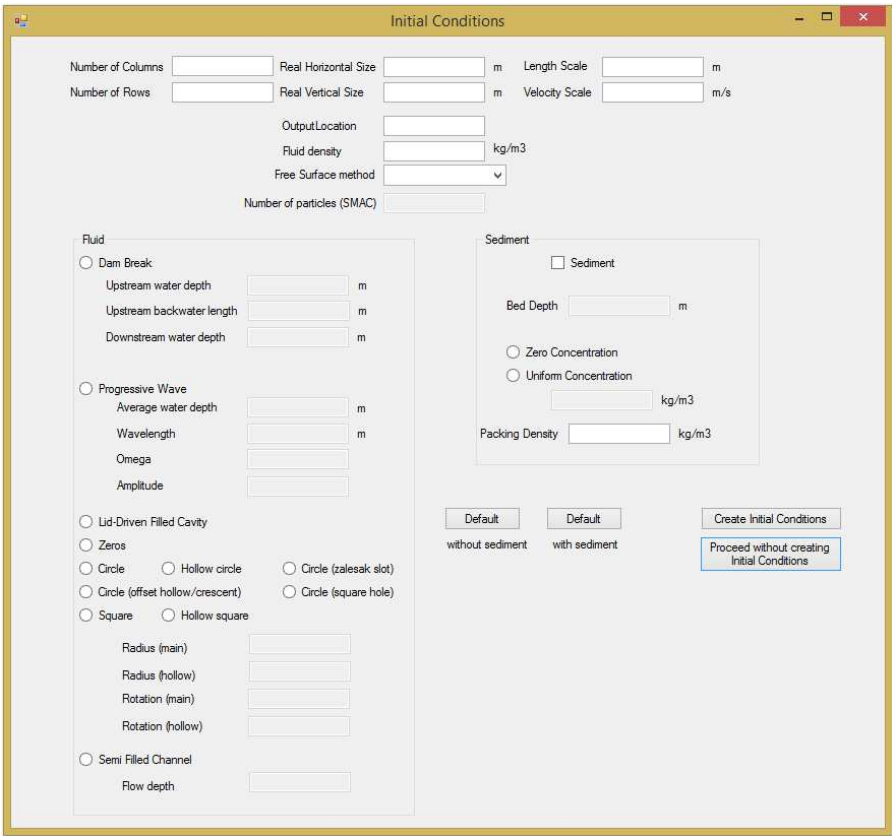


Figure 88: User interface, panel 1. Initial conditions files.

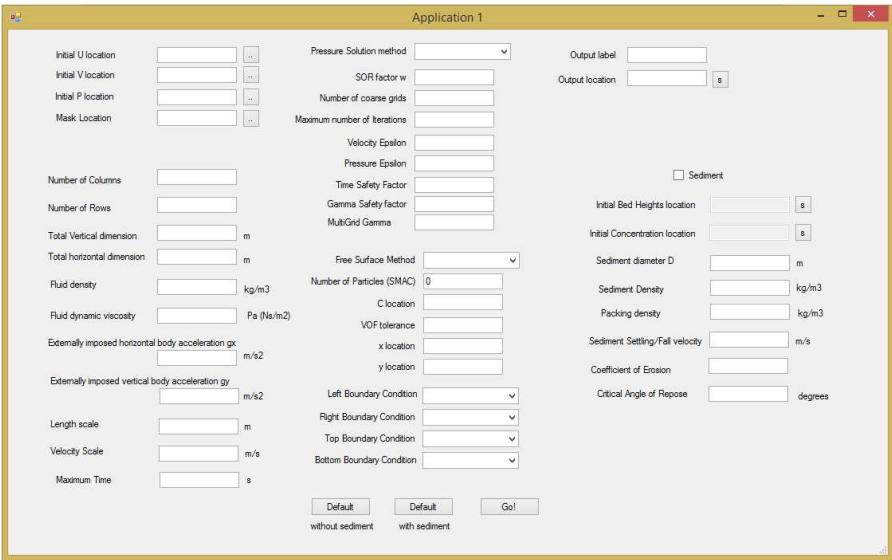


Figure 89: User interface, panel 2. Argument Pass.

ALGORITHMS

All of the algorithms presented below are given in pseudocode to allow for full understanding without prior knowledge of any specific programming language. All code was written from scratch, imperatively in C++, within an object-orientated structure.

B.1 NAVIER-STOKES SOLUTION

The algorithms presented below represent the procedure used for the solution of the Navier-Stokes Equations.

Algorithm 5 : Main Procedure

```

Input :
Output : u, v, p, c, C, Conservation floats
1 Set Initial Conditions from reading input files and passed values;
2 Set preliminary guess values for F and G with the values from u
  and v respectively;
3 while do
4   Assign Dt and Gamma;
5   Run Navier Stokes Solver for pressure;
6   Update velocity values with newly calculated pressures;
7   Impose boundary conditions, including free surface
    conditions if a free surface exists and variable bed conditions
    if a sediment bed exists;
8   if Volume is not full or sediment is being tracked then
9     Run freesurface/sediment solver;
10    Run Flotsam/Jetsam remover;
11    Update flags;
12    Zero appropriate empty cells;
13    Re-impose boundary conditions;
14  end
15  Increase time by  $\delta t$  and iterate the time step counter.;
16  if Time step counter is a multiple of a given value then
17    Print values to file;
18  end
19 end

```

B.2 FREE SURFACE SOLUTION

The algorithms presented below represent the procedure for determination of the free surface location and the movement of the free surface from one time step to the next.

Algorithm 6 : SLIC volume fraction Tracking method (one fluid), x-sweep

Input : Volume fraction field, c , horizontal component velocity field, u

Output : Outward volume flux fields, F_l and F_r

```

1  forall the Fluid Cells do
2      if Cell is Full ( $C = 1$ ) then
3          Calculate outward flux values, Full Cells (x-sweep:
              algorithm 14, y-sweep: algorithm 15);
4      else if Cell contains a surface ( $0 < c < 1$ ) then
5          if  $0 < c_{i,j} < 1$  then
6              if  $c_{i-1,j} > 0$  and  $c_{i+1,j} > 0$  then
7                  Calculate outward flux values, H-cell (algorithm
                      18);
8              else if  $c_{i-1,j} > 0$  and  $c_{i+1,j} = 0$  then
9                  Calculate outward flux values, V-cell (algorithm
                      16);
10             else if  $c_{i-1,j} = 0$  and  $c_{i+1,j} > 0$  then
11                 Calculate outward flux values, V-cell (algorithm
                      16);
12             else if  $c_{i-1,j} = 0$  and  $c_{i+1,j} = 0$  then
13                 Calculate outward flux values, Sandwich-cell
                      (algorithm 24);
14             end
15         end
16 end

```

Algorithm 7 : SLIC volume fraction Tracking method (one fluid), y-sweep

Input : Volume fraction field, c , vertical component velocity field, v

Output : Outward volume flux fields, F_t and F_b

```

1 forall the Fluid Cells do
2   if Cell is Full ( $C = 1$ ) then
3     Calculate outward flux values, Full Cells (x-sweep:
      algorithm 14, y-sweep: algorithm 15);
4   else if Cell contains a surface ( $0 < c < 1$ ) then
5     if  $0 < c_{i,j} < 1$  then
6       if  $c_{i,j-1} > 0$  and  $c_{i,j+1} > 0$  then
7         Calculate outward flux values, V-cell (algorithm
          17);
8       else if  $c_{i,j-1} > 0$  and  $c_{i,j+1} = 0$  then
9         Calculate outward flux values, H-cell (algorithm
          19);
10      else if  $c_{i,j-1} = 0$  and  $c_{i,j+1} > 0$  then
11        Calculate outward flux values, H-cell (algorithm
          19);
12      else if  $c_{i,j-1} = 0$  and  $c_{i,j+1} = 0$  then
13        Horizontal Sandwich;
14        Calculate outward flux values, Sandwich-cell
          (algorithm 30);
15    end
16  end
17 end

```

Algorithm 8 : Chorin's FLAP volume fraction Tracking method (one fluid), x-sweep, Part 1

Input : Volume fraction field, c , horizontal component velocity field, u

Output : Outward volume flux fields, F_l and F_r

```

1 for all Fluid Cells do
2   if Cell is Full ( $C = 1$ ) then
3     Calculate outward flux values, Full Cells (x-sweep:
      algorithm 14, y-sweep: algorithm 15);
4   else if  $0 < c_{i-1,j} < 1$  and  $0 < c_{i+1,j} < 1$  then
5     Calculate outward flux values, H-cell (algorithm 18);
6   else if  $c_{i-1,j} > 0$  and  $c_{i+1,j} = 0$  then
7     if  $(c_{i,j+1} > 0 \text{ and } c_{i,j-1} > 0)$  or  $(c_{i,j+1} = 0 \text{ and } c_{i,j-1} = 0)$ 
      then
8       Calculate outward flux values, VI-cell (algorithm 16);
9     else if  $c_{i,j+1} = 0$  and  $c_{i,j-1} > 0$  then
10      Calculate outward flux values, BLB-cell (algorithm 20);
11    else if  $c_{i,j+1} > 0$  and  $c_{i,j-1} = 0$  then
12      Calculate outward flux values, TLB-cell (algorithm 20);
13    end
14  else if  $c_{i-1,j} = 0$  and  $c_{i+1,j} > 0$  then
15    if  $(c_{i,j+1} > 0 \text{ and } c_{i,j-1} > 0)$  or  $(c_{i,j+1} = 0 \text{ and } c_{i,j-1} = 0)$ 
      then
16      Calculate outward flux values, Vr-cell (algorithm 16);
17    else if  $c_{i,j+1} = 0$  and  $c_{i,j-1} > 0$  then
18      Calculate outward flux values, BRB-cell (algorithm 21);
19    else if  $c_{i,j+1} > 0$  and  $c_{i,j-1} = 0$  then
20      Calculate outward flux values, TRB-cell (algorithm 21);
21    end

```

Algorithm 9 : Chorin's FLAP volume fraction Tracking method
 (one fluid), x-sweep, Part 2

```

22
23   else if  $c_{i-1,j} = 1$  and  $c_{i+1,j} < 1$  then
24       if  $(c_{i,j+1} = 1$  and  $c_{i,j-1} = 1)$  or  $(c_{i,j+1} < 1$  and  $c_{i,j-1} < 1)$ 
25           then
26               Calculate outward flux values, VI-cell (algorithm 16);
27           else if  $c_{i,j+1} = 1$  and  $c_{i,j-1} < 1$  then
28               Calculate outward flux values, BLW-cell (algorithm
29                   23);
30           else if  $c_{i,j+1} < 1$  and  $c_{i,j-1} = 1$  then
31               Calculate outward flux values, TLW-cell (algorithm
32                   23);
33           end
34   else if  $c_{i-1,j} < 1$  and  $c_{i+1,j} = 1$  then
35       if  $(c_{i,j+1} = 1$  and  $c_{i,j-1} = 1)$  or  $(c_{i,j+1} < 1$  and  $c_{i,j-1} < 1)$ 
36           then
37               Calculate outward flux values, Vr-cell (algorithm 16);
38           else if  $c_{i,j+1} = 1$  and  $c_{i,j-1} < 1$  then
39               Calculate outward flux values, BRW-cell (algorithm
40                   22);
41           else if  $c_{i,j+1} < 1$  and  $c_{i,j-1} = 1$  then
42               Calculate outward flux values, TRW-cell (algorithm
43                   22);
44           end
45   else if  $c_{i-1,j} = 1$  and  $c_{i+1,j} = 1$  then
46       Calculate outward flux values, WF-cell (algorithm 25);
47   else if  $c_{i-1,j} = 0$  and  $c_{i+1,j} = 0$  then
48       Calculate outward flux values, BF-cell (algorithm 24);
49   end
50 end

```

Algorithm 10 : Chorin's FLAP volume fraction Tracking method
(one fluid), y-sweep, Part 1

Input : Volume fraction field, c , vertical component velocity field,
 v

Output : Outward volume flux fields, F_{lt} and F_b

```

1 for all Fluid Cells do
2   if Cell is Full ( $C = 1$ ) then
3     Calculate outward flux values, Full Cells (x-sweep:
      algorithm 14, y-sweep: algorithm 15);
4   else if  $0 < c_{i,j-1} < 1$  and  $0 < c_{i,j+1} < 1$  then
5     Calculate outward flux values, V-cell (algorithm 17);
6   else if  $c_{i,j-1} > 0$  and  $c_{i,j+1} = 0$  then
7     if  $(c_{i+1,j} > 0 \text{ and } c_{i-1,j} > 0)$  or  $(c_{i+1,j} = 0 \text{ and } c_{i-1,j} = 0)$ 
      then
8       Calculate outward flux values, Hb-cell (algorithm 19);
9     else if  $c_{i+1,j} = 0$  and  $c_{i-1,j} > 0$  then
10      Calculate outward flux values, BLB-cell (algorithm 26);
11    else if  $c_{i+1,j} > 0$  and  $c_{i-1,j} = 0$  then
12      Calculate outward flux values, BRB-cell (algorithm 26);
13    end
14  else if  $c_{i,j-1} = 0$  and  $c_{i,j+1} > 0$  then
15    if  $(c_{i+1,j} > 0 \text{ and } c_{i-1,j} > 0)$  or  $(c_{i+1,j} = 0 \text{ and } c_{i-1,j} = 0)$ 
      then
16      Calculate outward flux values, Ht-cell (algorithm 19);
17    else if  $c_{i+1,j} = 0$  and  $c_{i-1,j} > 0$  then
18      Calculate outward flux values, TLB-cell (algorithm 27);
19    else if  $c_{i+1,j} > 0$  and  $c_{i-1,j} = 0$  then
20      Calculate outward flux values, TRB-cell (algorithm 27);
21    end

```

Algorithm 11 : Chorin's FLAP volume fraction Tracking method
 (one fluid), y-sweep, Part 2

```

22
23   else if  $c_{i,j-1} = 1$  and  $c_{i,j+1} < 1$  then
24       if  $(c_{i,j+1} = 1$  and  $c_{i,j-1} = 1)$  or  $(c_{i,j+1} < 1$  and  $c_{i,j-1} < 1)$ 
25           then
26               Calculate outward flux values, Hb-cell (algorithm 19);
27           else if  $c_{i,j+1} = 1$  and  $c_{i,j-1} < 1$  then
28               Calculate outward flux values, TLW-cell (algorithm
29                   28);
30           else if  $c_{i,j+1} < 1$  and  $c_{i,j-1} = 1$  then
31               Calculate outward flux values, TRW-cell (algorithm
32                   28);
33           end
34   else if  $c_{i,j-1} < 1$  and  $c_{i,j+1} = 1$  then
35       if  $(c_{i+1,j} = 1$  and  $c_{i-1,j} = 1)$  or  $(c_{i+1,j} < 1$  and  $c_{i-1,j} < 1)$ 
36           then
37               Calculate outward flux values, Ht-cell (algorithm 17);
38           else if  $c_{i+1,j} = 1$  and  $c_{i-1,j} < 1$  then
39               Calculate outward flux values, BLW-cell (algorithm
40                   29);
41           else if  $c_{i+1,j} < 1$  and  $c_{i-1,j} = 1$  then
42               Calculate outward flux values, BRW-cell (algorithm
43                   29);
44           end
45   else if  $c_{i,j-1} = 1$  and  $c_{i,j+1} = 1$  then
46       Calculate outward flux values, HWF-cell (algorithm 30);
47   else if  $c_{i,j-1} = 0$  and  $c_{i,j+1} = 0$  then
48       Calculate outward flux values, HBF-cell (algorithm 30);
49   end
50 end

```

Algorithm 12 : Young's Volume fraction Tracking method

Input : Volume fraction field, c , horizontal component velocity field, u

Output : Outward volume flux fields, F_l and F_r

```

1  forall the Fluid Cells do
2      if Cell is Full ( $C = 1$ ) then
3          Calculate outward flux values, Full Cells (x-sweep:
              algorithm 14, y-sweep: algorithm 15);
4      else if Cell contains a surface ( $0 < c < 1$ ) then
5          if Surface is Vertical then
6              Calculate outward flux values, V-cell (x-sweep:
              algorithm 16, y-sweep: algorithm 17);
7          else if Surface is Horizontal then
8              Calculate outward flux values, H-cell (x-sweep:
              algorithm 18, y-sweep: algorithm 19);
9          else
10             Calculate Surface angles  $\alpha$  and  $\beta$ ;
11             if  $\alpha > 0$  and  $\alpha < \frac{\pi}{2}$  then
12                 Calculate cell edge intersection points (algorithm
                    13);
13                 if  $Stencil_R > Stencil_L$  or  $Stencil_B > Stencil_T$  then
14                     Calculate outward flux values, A cell (x-sweep:
                        algorithms 31 & 32, y-sweep: algorithms 33 &
                        34);
15                 else
16                     Rotate Cell  $180^\circ$ ;
17                     Calculate outward flux values, C cell (x-sweep:
                        algorithms 39 & 40, y-sweep: algorithms 41 &
                        42);
18                 end
19             else if  $\alpha < 0$  and  $\alpha > \frac{-\pi}{2}$  then
20                 Rotate cell  $90^\circ$ ,  $\alpha = \frac{\pi}{2} - |\alpha|$  and  $\beta = |\beta|$ ;
21                 Calculate cell edge intersection points;
22                 Rotate Cell back;
23                 if  $Stencil_R > Stencil_L$  or  $Stencil_T > Stencil_B$  then
24                     Calculate outward flux values, D cell (x-sweep:
                        algorithms 43 & 44, y-sweep: algorithms 45 &
                        46);
25                 else
26                     Calculate outward flux values, B cell (x-sweep:
                        algorithms 35 & 36, y-sweep: algorithms 37 &
                        38);
27                 end
28             end
29         end
30     end
31 end

```

Algorithm 13 : Edge Intersection Calculator

Input : Interface angle α , cell volume fraction $c_{i,j}$ **Output** : Edge intersections in the form of the filled proportion of entire edge length, s_T , s_R , s_B and s_L

```

1  if  $\alpha < \frac{\pi}{4}$  then
2      if  $c \leq 0.5 \tan(\alpha)$  then
3          Case I: Set top edge filled proportion,  $s_T$ , to 0;
4          Set right edge filled proportion,  $s_R$ , to  $\sqrt{2c_{i,j} \tan(\alpha)}$ ;
5          Set bottom edge filled proportion,  $s_B$ , to  $\sqrt{2c_{i,j} \cot(\alpha)}$ ;
6          Set left edge filled proportion,  $s_L$ , to 0;
7      else if  $c \leq (1 - 0.5 \tan(\alpha))$  then
8          Case II: Set  $s_T$ , to 0;
9          Set  $s_R$ , to  $c_{i,j} + 0.5 \tan(\alpha)$ ;
10         Set  $s_B$ , to 1;
11         Set  $s_L$ , to  $c_{i,j} - 0.5 \tan(\alpha)$ ;
12     else
13         Case IV: Set  $s_T$ , to  $1 - \sqrt{2(1 - c_{i,j}) \cot(\alpha)}$ ;
14         Set  $s_R$ , to 1;
15         Set  $s_B$ , to 1;
16         Set  $s_L$ , to  $1 - \sqrt{2(1 - c_{i,j}) \tan(\alpha)}$ ;
17     end
18 else
19     if  $c \leq 0.5 \cot(\alpha)$  then
20         Case I: Set  $s_T$ , to 0;
21         Set  $s_R$ , to  $\sqrt{2c_{i,j} \tan(\alpha)}$ ;
22         Set  $s_B$ , to  $\sqrt{2c_{i,j} (\cot(\alpha))}$ ;
23         Set  $s_L$ , to 0;
24     else if  $c \leq (1 - 0.5 \cot(\alpha))$  then
25         Case III: Set  $s_T$ , to 0;
26         Set  $s_R$ , to  $c_{i,j} - 0.5 \cot(\alpha)$ ;
27         Set  $s_B$ , to 1;
28         Set  $s_L$ , to  $c_{i,j} + 0.5 \cot(\alpha)$ ;
29     else
30         Case IV: Set  $s_T$ , to  $1 - \sqrt{2(1 - c_{i,j}) \cot(\alpha)}$ ;
31         Set  $s_R$ , to 1;
32         Set  $s_B$ , to 1;
33         Set  $s_L$ , to  $1 - \sqrt{2(1 - c_{i,j}) \tan(\alpha)}$ ;
34     end
35 end

```

Algorithm 14 : Flux Calculator, x-sweep, Full Cells

Input : Horizontal component velocity field, u **Output** : Outward volume flux values for full cells, Fl and Fr

```

1 if  $u_{i,j} > 0$  then
2   | Set outward flux from cell  $(i,j)$ 's right edge,  $Fr_{i,j}$ , to  $u_{i,j}\delta t\delta y$ ;
3 end
4 if  $u_{i-1,j} < 0$  then
5   | Set outward flux from cell  $(i,j)$ 's left edge,  $Fl_{i,j}$ , to  $|u_{i-1,j}|\delta t\delta y$ ;
6 end

```

Algorithm 15 : Flux Calculator, y-sweep, Full Cells

Input : Vertical component velocity field, u **Output** : Outward volume flux values for full cells, Ft and Fb

```

1 if  $v_{i,j} > 0$  then
2   | Set outward flux from cell  $(i,j)$ 's top edge,  $Ft_{i,j}$ , to  $v_{i,j}\delta t\delta x$ ;
3 end
4 if  $v_{i-1,j} < 0$  then
5   | Set outward flux from cell  $(i,j)$ 's bottom edge,  $Fb_{i,j}$ , to
    |  $|v_{i-1,j}|\delta t\delta x$ ;
6 end

```

Algorithm 16 : Flux Calculator, x-sweep, V-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u

Output : Outward volume flux values for vertical surface cells, Fl and Fr

```

1 if Fluid is on the left then
2   if  $u_{i,j} > 0$  and  $u_{i,j}\delta t > (1 - c_{i,j})\delta x$  then
3     Set outward flux from cell  $(i,j)$ 's right edge,  $Fr_{i,j}$ , to
       $(u_{i,j}\delta t - (1 - c_{i,j})\delta x)\delta y$ ;
4   end
5   if  $u_{i-1,j} < 0$  and  $|u_{i-1,j}|\delta t < c_{i,j}\delta x$  then
6     Set outward flux from cell  $(i,j)$ 's left edge,  $Fl_{i,j}$ , to
       $|u_{i-1,j}|\delta t\delta y$ ;
7   else
8     Set outward flux from cell  $(i,j)$ 's left edge,  $Fl_{i,j}$ , to  $c_{i,j}\delta x\delta y$ ;
9   end
10 else
11   if  $u_{i,j} > 0$  and  $u_{i,j}\delta t < c_{i,j}\delta x$  then
12     Set outward flux from cell  $(i,j)$ 's right edge,  $Fr_{i,j}$ , to
       $u_{i,j}\delta t\delta y$ ;
13   else
14     Set outward flux from cell  $(i,j)$ 's right edge,  $Fr_{i,j}$ , to
       $c_{i,j}\delta x\delta y$ ;
15   end
16   if  $u_{i-1,j} < 0$  and  $|u_{i-1,j}|\delta t > (1 - c_{i,j})\delta x$  then
17     Set outward flux from cell  $(i,j)$ 's left edge,  $Fl_{i,j}$ , to
       $(|u_{i-1,j}|\delta t - (1 - c_{i,j})\delta x)\delta y$ ;
18   end
19 end

```

Algorithm 17 : Flux Calculator, y-sweep, V-Cells

Input : Volume fraction field, c , Horizontal component velocity field, v

Output : Outward volume flux values for vertical surface cells, F_t and F_b

```

1 if Fluid is on the left or Right then
2   if  $v_{i,j} > 0$  then
3     Set outward flux from cell  $(i,j)$ s top edge,  $F_{t,i,j}$ , to
       $v_{i,j} \delta t c_{i,j} \delta x$ ;
4   end
5   if  $v_{i-1,j} < 0$  then
6     Set outward flux from cell  $(i,j)$ s bottom edge,  $F_{b,i,j}$ , to
       $|v_{i,j}| \delta t c_{i,j} \delta x$ ;
7   end
8 end

```

Algorithm 18 : Flux Calculator, x-sweep, H-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u

Output : Outward volume flux values for vertical surface cells, F_l and F_r

```

1 if  $u_{i,j} > 0$  then
2   Set outward flux from cell  $(i,j)$ s right edge,  $F_{r,i,j}$ , to
       $u_{i,j} \delta t c_{i,j} \delta y$ ;
3 end
4 if  $u_{i-1,j} < 0$  then
5   Set outward flux from cell  $(i,j)$ s left edge,  $F_{l,i,j}$ , to
       $|u_{i-1,j}| \delta t c_{i-1,j} \delta y$ ;
6 end

```

Algorithm 19 : Flux Calculator, y-sweep, H-Cells

Input : Volume fraction field, c , Horizontal component velocity field, v

Output : Outward volume flux values for vertical surface cells, F_t and F_b

```

1  if Fluid is below then
2      if  $v_{i,j} > 0$  and  $v_{i,j}\delta t > (1 - c_{i,j})\delta y$  then
3          Set outward flux from cell  $(i,j)$ s top edge,  $F_{t,i,j}$ , to
             $(v_{i,j}\delta t - (1 - c_{i,j})\delta y)\delta x$ ;
4      end
5      if  $v_{i,j-1} < 0$  and  $|v_{i,j-1}|\delta t < c_{i,j}\delta y$  then
6          Set outward flux from cell  $(i,j)$ s bottom edge,  $F_{b,i,j}$ , to
             $|v_{i,j-1}|\delta t\delta x$ ;
7      else
8          Set outward flux from cell  $(i,j)$ s bottom edge,  $F_{b,i,j}$ , to
             $c_{i,j}\delta x\delta y$ ;
9      end
10 else
11     if  $v_{i,j} > 0$  and  $v_{i,j}\delta t < c_{i,j}\delta y$  then
12         Set outward flux from cell  $(i,j)$ s top edge,  $F_{t,i,j}$ , to  $v_{i,j}\delta t\delta x$ ;
13     else
14         Set outward flux from cell  $(i,j)$ s top edge,  $F_{t,i,j}$ , to  $c_{i,j}\delta x\delta y$ ;
15     end
16     if  $v_{i,j-1} < 0$  and  $|v_{i,j-1}|\delta t > (1 - c_{i,j})\delta y$  then
17         Set outward flux from cell  $(i,j)$ s bottom edge,  $F_{b,i,j}$ , to
             $(|v_{i,j-1}|\delta t - (1 - c_{i,j})\delta y)\delta x$ ;
18     end
19 end
  
```

Algorithm 20 : Flux Calculator, x-sweep, BL-B or TL-B-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u

Output : Outward volume flux values for vertical surface cells, Fl

```

1   $sbt = \sqrt{\frac{c_{i,j}c_{i,j-1}}{c_{i-1,j}}}$ ; and  $Fr$ 
2   $sl = \frac{c_{i,j}}{sbt}$ ;
3  if  $sl > 1$  then
4  |    $sl = 1$ ;
5  |    $sb = st = sbt = c_{i,j}$ ;
6  else if  $sbt > 1$  then
7  |    $sbt = 1$ ;
8  |    $sl = sr = c_{i,j}$ ;
9  end
10 if  $u_{i,j} > 0$  then
11 |   if  $u_{i,j}\delta t > (1 - sbt)\delta x$  then
12 | |    $Fr_{i,j} = (u_{i,j}\delta t - (1 - sbt)\delta x)sl\delta y$ ;
13 |   end
14 |   if  $u_{i-1,j} < 0$  then
15 | |   if  $|u_{i-1,j}|\delta t < sbt\delta x$  then
16 | | |    $Fl_{i,j} = |u_{i-1,j}|\delta t sl\delta y$ ;
17 | |   else
18 | | |    $Fl_{i,j} = c_{i,j}\delta x\delta y$ ;
19 | |   end
20 |   end
21 end

```

Algorithm 21 : Flux Calculator, x-sweep, BR-B or TR-B-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u

Output : Outward volume flux values for vertical surface cells, Fl

```

1   $sbt = \sqrt{\frac{c_{i,j}c_{i,j-1}}{c_{i-1,j}}}$ ;
2   $sr = \frac{c_{i,j}}{sbt}$ ;
3  if  $sr > 1$  then
4       $sr = 1$ ;
5       $sb = st = sbt = c_{i,j}$ ;
6  else if  $sbt > 1$  then
7       $sbt = 1$ ;
8       $sl = sr = c_{i,j}$ ;
9  end
10 if  $u_{i,j} > 0$  then
11     if  $u_{i,j}\delta t < sbt\delta x$  then
12          $Fr_{i,j} = u_{i,j}\delta t sr\delta y$ ;
13     else
14          $Fr_{i,j} = c_{i,j}\delta x\delta y$ ;
15     end
16     if  $u_{i-1,j} < 0$  then
17         if  $|u_{i-1,j}|\delta t > (1 - sbt)\delta x$  then
18              $Fl_{i,j} = (|u_{i-1,j}|\delta t - (1 - sbt)\delta x)sr\delta y$ ;
19         end
20     end
21 end

```

Algorithm 22 : Flux Calculator, x-sweep, BR-W or TR-W-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u **Output** : Outward volume flux values for vertical surface cells, Fl

and Fr

```

1   $sbt = 1 - \sqrt{\frac{(1-c_{i,j})(1-c_{i,j}-1)}{(1-c_{i+1,j})}};$ 
2   $sr = 1 - \frac{(1-c_{i,j})}{(1-sbt)};$ 
3  if  $sr < 0$  then
4  |    $sr = 0;$ 
5  |    $sb = st = sbt = c_{i,j};$ 
6  else if  $sbt < 0$  then
7  |    $sbt = 0;$ 
8  |    $sl = sr = c_{i,j};$ 
9  end
10 if  $u_{i,j} > 0$  then
11 |   if  $u_{i,j}\delta t < (1-sbt)\delta x$  then
12 | |    $Fr_{i,j} = u_{i,j}\delta t sr\delta y;$ 
13 |   else
14 | |    $Fr_{i,j} = sr\delta y(1-sbt)\delta x + (u_{i,j}\delta t - (1-sbt)\delta x)\delta y;$ 
15 |   end
16 end
17 if  $u_{i-1,j} < 0$  then
18 |   if  $|u_{i-1,j}|\delta t < sbt\delta x$  then
19 | |    $Fl_{i,j} = (|u_{i-1,j}|\delta t - (1-sbt)\delta x)sr\delta y;$ 
20 |   end
21 |   else
22 | |    $Fl_{i,j} = st\delta x\delta y + (|u_{i-1,j}|\delta t - st\delta x)sr\delta y$ 
23 |   end
24 end

```

Algorithm 23 : Flux Calculator, x-sweep, BL-W or TL-W-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u

Output : Outward volume flux values for vertical surface cells, Fl and Fr

```

1   $sbt = 1 - \sqrt{\frac{(1-c_{i,j})(1-c_{i,j-1})}{(1-c_{i+1,j})}}$ ;
2   $sl = 1 - \frac{(1-c_{i,j})}{(1-sbt)}$ ;
3  if  $sl < 0$  then
4       $sr = 0$ ;
5       $sb = st = sbt = c_{i,j}$ ;
6  else if  $sbt < 0$  then
7       $sbt = 0$ ;
8       $sl = sr = c_{i,j}$ ;
9  end
10 if  $u_{i,j} > 0$  then
11     if  $u_{i,j}\delta t < sbt\delta x$  then
12          $Fr_{i,j} = u_{i,j}\delta t\delta y$ ;
13     else
14          $Fr_{i,j} = sbt\delta x\delta y + (u_{i,j}\delta t - sbt\delta x)sl\delta y$ 
15     end
16 end
17 if  $u_{i-1,j} < 0$  then
18     if  $|u_{i-1,j}|\delta t < (1-sbt)\delta x$  then
19          $Fl_{i,j} = (|u_{i-1,j}|\delta t)sl\delta y$ ;
20     end
21     else
22          $Fl_{i,j} = sr\delta y(1-sbt)\delta x + (|u_{i-1,j}|\delta t - (1-sbt)\delta x)\delta y$ ;
23     end
24 end

```

Algorithm 24 : Flux Calculator, x-sweep, BF-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u

Output : Outward volume flux values for vertical surface cells, Fl and Fr

```

1  if  $u_{i,j} > 0$  then
2    if  $u_{i,j}\delta t < 0.5(1 - c_{i,j})\delta x$  then
3       $Fr_{i,j} = 0;$ 
4    else if  $u_{i,j}\delta t < 0.5(1 - c_{i,j})\delta x + c_{i,j}\delta x$  then
5       $Fr_{i,j} = (u_{i,j}\delta t - 0.5(1 - c_{i,j})\delta x)\delta y;$ 
6    else
7       $Fr_{i,j} = c_{i,j}\delta x\delta y;$ 
8    end
9  end
10 if  $u_{i-1,j} < 0$  then
11   if  $|u_{i-1,j}|\delta t < 0.5(1 - c_{i,j})\delta x$  then
12      $Fl_{i,j} = 0;$ 
13   end
14   else if  $|u_{i-1,j}|\delta t < 0.5(1 - c_{i,j})\delta x + c_{i,j}\delta x$  then
15      $Fl_{i,j} = (|u_{i-1,j}|\delta t - 0.5(1 - c_{i,j})\delta x)\delta y;$ 
16   else
17      $Fl_{i,j} = c_{i,j}\delta x\delta y;$ 
18   end
19 end

```

Algorithm 25 : Flux Calculator, x-sweep, WF-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u

Output : Outward volume flux values for vertical surface cells, Fl and Fr

```

1  if  $u_{i,j} > 0$  then
2      if  $u_{i,j}\delta t < 0.5c_{i,j}\delta x$  then
3           $Fr_{i,j} = u_{i,j}\delta t\delta y;$ 
4      else if  $u_{i,j}\delta t < 0.5c_{i,j}\delta x + (1 - st)\delta x$  then
5           $Fr_{i,j} = 0.5c_{i,j}\delta x\delta y;$ 
6      else
7           $Fr_{i,j} = 0.5c_{i,j}\delta x\delta y + (u_{i,j}\delta t - (0.5c_{i,j}\delta x + (1 - st)\delta x))\delta y;$ 
8      end
9  end
10 if  $u_{i-1,j} < 0$  then
11     if  $|u_{i-1,j}|\delta t < 0.5c_{i,j}\delta x$  then
12          $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y;$ 
13     end
14     else if  $|u_{i-1,j}|\delta t < 0.5c_{i,j}\delta x + (1 - c_{i,j})\delta x$  then
15          $Fl_{i,j} = 0.5c_{i,j}\delta x\delta y;$ 
16     else
17          $Fl_{i,j} = 0.5c_{i,j}\delta x\delta y + (|u_{i-1,j}|\delta t - (0.5c_{i,j}\delta x + (1 - c_{i,j})\delta x))\delta y;$ 
18     end
19 end

```

Algorithm 26 : Flux Calculator, y-sweep, BL-B or BR-B-Cells

Input : Volume fraction field, c , Vertical component velocity field,
 v

Output : Outward volume flux values for vertical surface cells,

Fb and Ft

```

1  sb =  $\sqrt{\frac{c_{i,j}c_{i,j-1}}{c_{i-1,j}}}$ ;
2  srl =  $\frac{c_{i,j}}{sb}$ ;
3  if srl > 1 then
4      |   srl = 1;
5      |   sb = st =  $c_{i,j}$ ;
6  else if sb > 1 then
7      |   sb = 1;
8      |   sl = sr = srl =  $c_{i,j}$ ;
9  end
10 if  $v_{i,j} > 0$  then
11     |   if  $v_{i,j}\delta t > (1 - srl)\delta y$  then
12         |       Fti,j =  $(v_{i,j}\delta t - (1 - srl)\delta y)sb\delta x$ ;
13     |   end
14     |   if  $v_{i,j-1} < 0$  then
15         |       if  $|v_{i,j-1}|\delta t < srl\delta y$  then
16             |           Fbi,j =  $|v_{i,j-1}|\delta tsb\delta x$ ;
17         |       else
18             |           Fbi,j =  $c_{i,j}\delta x\delta y$ ;
19         |       end
20     |   end
21 end

```

Algorithm 27 : Flux Calculator, y-sweep, TL-B or TR-B-Cells

Input : Volume fraction field, c , Vertical component velocity field,
 v

Output : Outward volume flux values for vertical surface cells,

Fb and Ft

```

1  st =  $\sqrt{\frac{c_{i,j}c_{i,j-1}}{c_{i-1,j}}}$ ;
2  srl =  $\frac{c_{i,j}}{st}$ ;
3  if srl > 1 then
4    srl = 1;
5    sb = st =  $c_{i,j}$ ;
6  else if st > 1 then
7    st = 1;
8    sl = sr = srl =  $c_{i,j}$ ;
9  end
10 if  $v_{i,j} > 0$  then
11   if  $v_{i,j}\delta t < srl\delta y$  then
12     Fti,j =  $v_{i,j}\delta t st\delta x$ ;
13   else
14     Fti,j =  $c_{i,j}\delta x\delta y$ ;
15   end
16   if  $v_{i,j-1} < 0$  then
17     if  $|v_{i,j-1}|\delta t > (1 - srl)\delta y$  then
18       Fbi,j =  $(|v_{i,j-1}|\delta t - (1 - srl)\delta y)st\delta x$ ;
19     end
20   end
21 end
  
```

Algorithm 28 : Flux Calculator, y-sweep, TL-W or TR-W-Cells

Input : Volume fraction field, c , Vertical component velocity field,
 v **Output** : Outward volume flux values for vertical surface cells,

Fb and Ft

```

1  st = 1 -  $\sqrt{\frac{(1-c_{i,j})(1-c_{i,j-1})}{(1-c_{i+1,j})}}$ ;
2  srl = 1 -  $\frac{(1-c_{i,j})}{(1-st)}$ ;
3  if srl < 0 then
4      |   srl = 0;
5      |   sb = st =  $c_{i,j}$ ;
6  else if st < 0 then
7      |   st = 0;
8      |   sl = sr = srl =  $c_{i,j}$ ;
9  end
10 if  $v_{i,j} > 0$  then
11     |   if  $v_{i,j}\delta t < (1 - srl)\delta y$  then
12         |       Fti,j =  $v_{i,j}\delta t st\delta x$ ;
13     else
14         |       Fti,j =  $st\delta x(1 - srl)\delta y + (v_{i,j}\delta t - (1 - srl)\delta y)\delta x$ ;
15     end
16 end
17 if  $v_{i,j-1} < 0$  then
18     |   if  $|v_{i,j-1}|\delta t < srl\delta y$  then
19         |       Fbi,j =  $(|v_{i,j-1}|\delta t)\delta x$ ;
20     end
21 else
22     |       Fbi,j =  $srl\delta x\delta y + (|v_{i,j-1}|\delta t - srl\delta y)st\delta x$ 
23     end
24 end

```

Algorithm 29 : Flux Calculator, y-sweep, BL-W or BR-W-Cells

Input : Volume fraction field, c , Vertical component velocity field,
 v **Output** : Outward volume flux values for vertical surface cells,

Fb and Ft

```

1  sb = 1 -  $\sqrt{\frac{(1-c_{i,j})(1-c_{i,j-1})}{(1-c_{i+1,j})}}$ ;
2  srl = 1 -  $\frac{(1-c_{i,j})}{(1-sb)}$ ;
3  if srl < 0 then
4      | srl = 0;
5      | sb = st =  $c_{i,j}$ ;
6  else if sbr < 0 then
7      | sb = 0;
8      | sl = sr = srl =  $c_{i,j}$ ;
9  end
10 if  $v_{i,j} > 0$  then
11     | if  $v_{i,j}\delta t < srl\delta y$  then
12         | Fti,j =  $v_{i,j}\delta t\delta x$ ;
13     else
14         | Fti,j =  $srl\delta x\delta y + (v_{i,j}\delta t - srl\delta y)sb\delta x$ 
15     end
16 end
17 if  $v_{i,j-1} < 0$  then
18     | if  $|v_{i,j-1}|\delta t < (1 - srl)\delta y$  then
19         | Fbi,j =  $(|v_{i,j-1}|\delta t)sb\delta x$ ;
20     end
21 else
22     | Fbi,j =  $sb\delta x(1 - srl)\delta y + (|v_{i-1,j}|\delta t - (1 - srl)\delta y)\delta x$ ;
23 end
24 end

```

Algorithm 30 : Flux Calculator, y-sweep, BF or WF-Cells

Input : Volume fraction field, c , Horizontal component velocity field, u **Output** : Outward volume flux values for vertical surface cells, Fl and Fr

```

1 if  $v_{i,j} > 0$  then
2   |  $Ft_{i,j} = v_{i,j} \delta t c_{i,j} \delta x$ ;
3 end
4 if  $v_{i,j-1} < 0$  then
5   |  $Fb_{i,j} = |v_{i,j-1}| \delta t c_{i,j} \delta x$ ;
6 end

```

Algorithm 31 : Flux Calculator, x-sweep, A-Cells, Part 1

Input : Volume fraction field, c , Horizontal component velocity field, u , Edge intersections, s_T , s_R , s_B and s_L **Output** : Outward volume flux values for A type surface cells, Fl and Fr

```

1 Determine case as in algorithm 13;
2 if Case I then
3   | if  $u_{i,j} > 0$  then
4     | if  $u_{i,j} \delta t \geq s_B \delta x$  then
5       |  $Fr_{i,j} = c_{i,j} \delta x \delta y$ ;
6     | else
7       |  $Fr_{i,j} = 0.5 u_{i,j} \delta t (2 - u_{i,j} \delta t / (s_B \delta x)) s_R \delta y$ ;
8     | end
9   | end
10  | if  $u_{i-1,j} < 0$  then
11    | if  $|u_{i-1,j}| \delta t \leq (1 - s_B) \delta x$  then
12      |  $Fl_{i,j} = 0$ ;
13    | else
14      |  $Fl_{i,j} = 0.5 (|u_{i-1,j}| \delta t - (1 - s_B) \delta x)^2 \tan(\beta)$ ;
15    | end
16  | end
17 else if Case II then
18   | if  $u_{i,j} > 0$  then
19     |  $Fr_{i,j} = u_{i,j} \delta t (s_R \delta y - 0.5 u_{i,j} \delta t \tan(\beta))$ ;
20   | end
21   | if  $u_{i-1,j} < 0$  then
22     |  $Fl_{i,j} = |u_{i-1,j}| \delta t (s_L \delta y + 0.5 |u_{i-1,j}| \delta t \tan(\beta))$ ;
23   | end

```

Algorithm 32 : Flux Calculator, x-sweep, A-Cells, Part 2

```

24 else if Case III then
25   if  $u_{i,j} > 0$  then
26     if  $u_{i,j}\delta t \leq s_T\delta x$  then
27        $Fr_{i,j} = u_{i,j}\delta t\delta y$ ;
28     else if  $u_{i,j}\delta t \leq s_B\delta x$  then
29        $Fr_{i,j} = u_{i,j}\delta t\delta y - 0.5(u_{i,j}\delta t - s_T\delta x)^2 \tan(\beta)$ ;
30     else
31        $Fr_{i,j} = c_{i,j}\delta x\delta y$ ;
32     end
33   end
34   if  $u_{i-1,j} < 0$  then
35     if  $u_{i-1,j}\delta t \leq (1 - s_B)\delta x$  then
36        $Fl_{i,j} = 0$ ;
37     else if  $u_{i-1,j}\delta t \leq (1 - s_T)\delta x$  then
38        $Fl_{i,j} = 0.5(|u_{i-1,j}|\delta t - (1 - s_B)\delta x)^2 \tan(\beta)$ ;
39     else
40        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y - (1 - c_{i,j})\delta x\delta y$ ;
41     end
42   end
43 else
44   if  $u_{i,j} > 0$  then
45     if  $u_{i,j}\delta t \leq s_T\delta x$  then
46        $Fr_{i,j} = u_{i,j}\delta t\delta y$ ;
47     else
48        $Fr_{i,j} = u_{i,j}\delta t\delta y - 0.5 \tan(\beta)(u_{i,j}\delta t - s_T\delta x)^2$ ;
49     end
50   end
51   if  $u_{i-1,j} < 0$  then
52     if  $|u_{i-1,j}|\delta t \geq (1 - s_T)\delta x$  then
53        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y - (1 - c_{i,j})\delta x\delta y$ ;
54     else
55        $Fl_{i,j} = |u_{i-1,j}|\delta t(s_L\delta y + 0.5|u_{i-1,j}|\delta t \tan(\beta))$ ;
56     end
57   end
58 end

```

Algorithm 33 : Flux Calculator, y-sweep, A-Cells, Part 1

Input : Volume fraction field, c , Vertical component velocity field, v , Edge intersections, s_T , s_R , s_B and s_L

Output : Outward volume flux values for A type surface cells, F_t and F_b

```

1 Determine case as in algorithm 13;
2 if Case I then
3   if  $v_{i,j} > 0$  and  $v_{i,j}\delta t > (1 - s_R)\delta y$  then
4      $F_{t,i,j} = 0.5(v_{i,j}\delta t - (1 - s_R)\delta y)^2(1/\tan(\beta));$ 
5   end
6   if  $v_{i,j-1} < 0$  then
7     if  $|v_{i,j-1}|\delta t \geq s_R\delta y$  then
8        $F_{b,i,j} = c_{i,j}\delta x\delta y;$ 
9     else
10       $F_{b,i,j} = 0.5|v_{i,j-1}|\delta t(2 - |v_{i,j-1}|\delta t/(s_R\delta y))s_B\delta x;$ 
11    end
12  end
13 else if Case II then
14   if  $v_{i,j} > 0$  then
15     if  $v_{i,j}\delta t \leq (1 - s_R)\delta y$  then
16        $F_{t,i,j} = 0$ 
17     else if  $v_{i,j}\delta t \leq (1 - s_L)\delta y$  then
18        $F_{t,i,j} = 0.5(v_{i,j}\delta t - (1 - s_R)\delta y)^2(1/\tan(\beta));$ 
19     else
20        $F_{t,i,j} = v_{i,j}\delta t\delta x - (1 - c_{i,j})\delta x\delta y;$ 
21     end
22   end
23   if  $v_{i,j-1} < 0$  then
24     if  $|v_{i,j-1}|\delta t \leq s_L\delta y$  then
25        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x$ 
26     end
27     else if  $|v_{i,j-1}|\delta t \leq s_R\delta y$  then
28        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x - 0.5(|v_{i,j-1}|\delta t - s_L\delta y)^2(1/\tan(\beta))$ 
29     else
30        $F_{b,i,j} = c_{i,j}\delta x\delta y$ 
31     end
32   end

```

Algorithm 34 : Flux Calculator, y-sweep, A-Cells, Part 2

```

33 else if Case III then
34   if  $v_{i,j} > 0$  then
35      $F_{t,i,j} = v_{i,j} \delta t (s_T \delta x + 0.5 v_{i,j} \delta t (1/\tan(\beta)))$ 
36   end
37   if  $v_{i,j-1} < 0$  then
38      $F_{b,i,j} = |v_{i,j-1}| \delta t (s_B \delta x - 0.5 |v_{i,j-1}| \delta t (1/\tan(\beta)))$ 
39   end
40 else
41   if  $v_{i,j} > 0$  then
42     if  $v_{i,j} \delta t \geq (1 - s_L) \delta y$  then
43        $F_{t,i,j} = v_{i,j} \delta t \delta x - (1 - c_{i,j}) \delta x \delta y;$ 
44     else
45        $F_{t,i,j} = v_{i,j} \delta t (s_T \delta x + 0.5 v_{i,j} \delta t (1/\tan(\beta)));$ 
46     end
47   end
48   if  $v_{i,j-1} < 0$  then
49     if  $|v_{i,j-1}| \delta t \leq s_L \delta y$  then
50        $F_{b,i,j} = |v_{i,j-1}| \delta t \delta x;$ 
51     else
52        $F_{b,i,j} = |v_{i,j-1}| \delta t \delta x - 0.5 (|v_{i,j-1}| \delta t - s_L \delta y)^2 (1/\tan(\beta));$ 
53     end
54   end
55 end

```

Algorithm 35 : Flux Calculator, x-sweep, B-Cells, Part 1

Input : Volume fraction field, c , Horizontal component velocity field, u , Edge intersections, s_T , s_R , s_B and s_L **Output :** Outward volume flux values for B type surface cells, Fl and Fr

```

1  if Case I then
2      if  $u_{i,j} > 0$  then
3          if  $|u_{i,j}|\delta t \leq (1 - s_B)\delta x$  then
4               $Fr_{i,j} = 0$ ;
5          else
6              Set outward flux from cell  $(i, j)$ s right edge,  $Fr_{i,j}$ , to
               $0.5(u_{i,j}\delta t - (1 - s_B)\delta x)^2 \tan(\beta)$ ;
7          end
8      end
9      if  $u_{i-1,j} < 0$  then
10         if  $|u_{i-1,j}|\delta t \geq s_B\delta x$  then
11              $Fl_{i,j} = c_{i,j}\delta x\delta y$ ;
12         else
13              $Fl_{i,j} = 0.5|u_{i-1,j}|\delta t(2 - |u_{i-1,j}|\delta t/(s_B\delta x))s_L\delta y$ ;
14         end
15     end
16 else if Case II then
17     Use Case III equations;
18     if  $u_{i,j} > 0$  then
19         if  $u_{i,j}\delta t \leq (1 - s_B)\delta x$  then
20              $Fr_{i,j} = 0$ ;
21         else if  $u_{i,j}\delta t \leq (1 - s_T)\delta x$  then
22              $Fr_{i,j} = 0.5(u_{i,j}\delta t - (1 - s_B)\delta x)^2 \tan(\beta)$ ;
23         else
24              $Fr_{i,j} = u_{i,j}\delta t\delta y - (1 - c_{i,j})\delta x\delta y$ ;
25         end
26     end
27     if  $u_{i-1,j} < 0$  then
28         if  $|u_{i-1,j}|\delta t \leq s_T\delta x$  then
29              $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y$ ;
30         else if  $|u_{i-1,j}|\delta t \leq s_B\delta x$  then
31              $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y - 0.5(|u_{i-1,j}|\delta t - s_T\delta x)^2 \tan(\beta)$ ;
32         else
33              $Fl_{i,j} = c_{i,j}\delta x\delta y$ ;
34         end
35     end

```

Algorithm 36 : Flux Calculator, x-sweep, B-Cells, Part 2

```

36 else if Case III then
37   Use Case II Equations;
38   if  $u_{i,j} > 0$  then
39      $Fr_{i,j} = u_{i,j}\delta t(s_R\delta y + 0.5 * u_{i,j}\delta t \tan(\beta));$ 
40   end
41   if  $u_{i-1,j} < 0$  then
42      $Fl_{i,j} = |u_{i-1,j}|\delta t(s_L\delta y - 0.5|u_{i-1,j}|\delta t \tan(\beta));$ 
43   end
44 else
45   if  $u_{i,j} > 0$  then
46     if  $u_{i,j}\delta t \geq (1 - s_T)\delta x$  then
47        $Fr_{i,j} = u_{i,j}\delta t\delta y - (1 - c_{i,j})\delta x\delta y;$ 
48     else
49        $Fr_{i,j} = u_{i,j}\delta t(s_R\delta y + 0.5u_{i,j}\delta t \tan(\beta));$ 
50     end
51   end
52   if  $u_{i-1,j} < 0$  then
53     if  $u_{i-1,j}\delta t \leq s_T\delta x$  then
54        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y;$ 
55     else
56        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y - 0.5 \tan(\beta)(|u_{i-1,j}|\delta t - s_T\delta x)^2;$ 
57     end
58   end
59 end

```

Algorithm 37 : Flux Calculator, y-sweep, B-Cells, Part 1

Input : Volume fraction field, c , Vertical component velocity field, v , Edge intersections, s_T , s_R , s_B and s_L

Output : Outward volume flux values for B type surface cells, F_t and F_b

```

1 if Case I then
2   if  $v_{i,j} > 0$  then
3     if  $|v_{i,j}|\delta t \leq (1 - s_L)\delta y$  then
4        $F_{t,i,j} = 0;$ 
5     else
6        $F_{t,i,j} = 0.5(v_{i,j}\delta t - (1.0 - s_L)\delta y)^2(1/\tan(\beta));$ 
7     end
8   end
9   if  $v_{i,j-1} < 0$  then
10    if  $|v_{i,j-1}|\delta t \geq s_L\delta y$  then
11       $F_{b,i,j} = c_{i,j}\delta x\delta y;$ 
12    else
13       $F_{b,i,j} = 0.5|v_{i,j-1}|\delta t(2 - |v_{i,j-1}|\delta t/(s_L\delta y))s_B\delta x;$ 
14    end
15  end
16 else if Case II then
17   Use Case III equations;
18   if  $v_{i,j} > 0$  then
19      $F_{t,i,j} = v_{i,j}\delta t(s_T\delta x + 0.5v_{i,j}\delta t(1/\tan(\beta)));$ 
20   end
21   if  $v_{i,j-1} < 0$  then
22      $F_{b,i,j} = |v_{i,j-1}|\delta t(s_B\delta x - 0.5|v_{i,j-1}|\delta t(1/\tan(\beta)));$ 
23   end

```

Algorithm 38 : Flux Calculator, y-sweep, B-Cells, Part 2

```

36 else if Case III then
37   Use Case II Equations;
38   if  $v_{i,j} > 0$  then
39     if  $v_{i,j}\delta t \leq (1 - s_L)\delta x$  then
40        $F_{t,i,j} = 0$ ;
41     else if  $v_{i,j}\delta t \leq (1 - s_R)\delta x$  then
42        $F_{t,i,j} = 0.5(v_{i,j}\delta t - (1.0 - S_L)\delta y)^2(1/\tan(\beta))$ ;
43     else
44        $F_{t,i,j} = v_{i,j}\delta t\delta x - (1 - c_{i,j})\delta x\delta y$ ;
45     end
46   end
47   if  $v_{i,j-1} < 0$  then
48     if  $|v_{i,j-1}|\delta t \leq s_R\delta x$  then
49        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x$ ;
50     else if  $|v_{i,j-1}|\delta t \leq s_L\delta x$  then
51        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x - 0.5(|v_{i,j-1}|\delta t - s_R\delta y)^2(1/\tan(\beta))$ ;
52     else
53        $F_{b,i,j} = c_{i,j}\delta x\delta y$ ;
54     end
55   end
56 else
57   if  $v_{i,j} > 0$  then
58     if  $v_{i,j}\delta t \geq (1 - s_R)\delta y$  then
59        $F_{t,i,j} = v_{i,j}\delta t\delta x - (1 - c_{i,j})\delta x\delta y$ ;
60     else
61        $F_{t,i,j} = v_{i,j}\delta t(s_T\delta x + 0.5v_{i,j}\delta t(1/\tan(\beta)))$ ;
62     end
63   end
64   if  $v_{i,j-1} < 0$  then
65     if  $|v_{i,j-1}|\delta t \leq s_R\delta y$  then
66        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x$ 
67     else
68        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x - (0.5(|v_{i,j-1}|\delta t - s_R\delta y)^2(1/\tan(\beta)))$ 
69     end
70   end
71 end

```

Algorithm 39 : Flux Calculator, x-sweep, C-Cells, Part 1

Input : Volume fraction field, c , Horizontal component velocity field, u , Edge intersections, s_T , s_R , s_B and s_L

Output : Outward volume flux values for A type surface cells, Fl and Fr

```

1 Determine case as in algorithm 13;
2 if Case I then
3   if  $u_{i,j} > 0$  then
4     if  $|u_{i,j}|\delta t \leq (1 - s_T)\delta x$  then
5        $Fr_{i,j} = 0$ ;
6     else
7        $Fr_{i,j} = 0.5(u_{i,j}\delta t - (1 - s_T)\delta x) \tan(\beta)$ ;
8     end
9   end
10  if  $u_{i-1,j} < 0$  then
11    if  $|u_{i-1,j}|\delta t \geq s_T\delta x$  then
12       $Fl_{i,j} = c_{i,j}\delta x\delta y$ ;
13    else
14       $Fl_{i,j} = 0.5|u_{i-1,j}|\delta t(2 - |u_{i-1,j}|\delta t/(s_T\delta x))s_L\delta y$ ;
15    end
16  end
17 else if Case II then
18   if  $u_{i,j} > 0$  then
19      $Fr_{i,j} = u_{i,j}\delta t(s_R\delta y + 0.5 * u_{i,j}\delta t \tan(\beta))$ ;
20   end
21   if  $u_{i-1,j} < 0$  then
22      $Fl_{i,j} = |u_{i-1,j}|\delta t(s_L\delta y - 0.5|u_{i-1,j}|\delta t \tan(\beta))$ ;
23   end

```

Algorithm 40 : Flux Calculator, x-sweep, C-Cells, Part 2

```

24 else if Case III then
25   if  $u_{i,j} > 0$  then
26     if  $u_{i,j}\delta t \leq (1 - s_T)\delta x$  then
27        $Fr_{i,j} = 0;$ 
28     else if  $u_{i,j}\delta t \leq (1 - s_B)\delta x$  then
29        $Fr_{i,j} = 0.5(u_{i,j}\delta t - (1 - s_T)\delta x)^2 \tan(\beta);$ 
30     else
31        $Fr_{i,j} = u_{i,j}\delta t\delta y - (1 - c_{i,j})\delta x\delta y;$ 
32     end
33   end
34   if  $u_{i-1,j} < 0$  then
35     if  $|u_{i-1,j}|\delta t \leq s_B\delta x$  then
36        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y;$ 
37     else if  $|u_{i-1,j}|\delta t \leq s_T\delta x$  then
38        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y - 0.5(|u_{i-1,j}|\delta t - s_B\delta x) \tan(\beta);$ 
39     else
40        $Fl_{i,j} = c_{i,j}\delta x\delta y;$ 
41     end
42   end
43 else
44   if  $u_{i,j} > 0$  then
45     if  $u_{i,j}\delta t \geq (1 - s_B)\delta x$  then
46        $Fr_{i,j} = u_{i,j}\delta t\delta y - (1 - c_{i,j})\delta x\delta y;$ 
47     else
48        $Fr_{i,j} = u_{i,j}\delta t(s_R\delta y + 0.5u_{i,j}\delta t \tan(\beta));$ 
49     end
50   end
51   if  $u_{i-1,j} < 0$  then
52     if  $|u_{i-1,j}|\delta t \leq s_B\delta x$  then
53        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y;$ 
54     else
55        $Fl_{i,j} = |u_{i-1,j}|\delta t\delta y - 0.5 \tan(\beta)(|u_{i-1,j}|\delta t - s_B\delta x)^2;$ 
56     end
57   end
58 end

```

Algorithm 41 : Flux Calculator, y-sweep, C-Cells, Part 1

Input : Volume fraction field, c , Horizontal component velocity field, u , Edge intersections, s_T , s_R , s_B and s_L

Output : Outward volume flux values for A type surface cells, F_l and F_r

```

1 Determine case as in algorithm 13;
2 if Case I then
3   if  $v_{i,j} > 0$  then
4     if  $v_{i,j}\delta t \geq s_L\delta x$  then
5        $F_{t,i,j} = c_{i,j}\delta x\delta y$ ;
6     else
7        $F_{t,i,j} = 0.5v_{i,j}\delta t(2 - v_{i,j}\delta t/(s_L\delta y))s_T\delta x$ ;
8     end
9   end
10  if  $v_{i,j-1} < 0$  then
11    if  $|v_{i,j-1}|\delta t \leq (1 - s_L)\delta x$  then
12       $F_{b,i,j} = 0$ ;
13    else
14       $F_{b,i,j} = 0.5(|v_{i,j-1}|\delta t - (1 - s_L)\delta y)^2(1/\tan(\beta))$ ;
15    end
16  end
17 else if Case II then
18   if  $v_{i,j} > 0$  then
19     if  $v_{i,j}\delta t \leq s_R\delta y$  then
20        $F_{t,i,j} = v_{i,j}\delta t\delta x$ ;
21     else if  $u_{i,j}\delta t \leq s_L\delta y$  then
22        $F_{t,i,j} = v_{i,j}\delta t\delta x - 0.5(v_{i,j}\delta t - s_R\delta y)^2(1/\tan(\beta))$ ;
23     else
24        $F_{t,i,j} = c_{i,j}\delta x\delta y$ ;
25     end
26   end
27   if  $v_{i,j-1} < 0$  then
28     if  $|v_{i,j-1}|\delta t \leq (1 - s_L)\delta y$  then
29        $F_{b,i,j} = 0$ ;
30     else if  $|v_{i,j-1}|\delta t \leq (1 - s_R)\delta y$  then
31        $F_{b,i,j} = 0.5(|v_{i,j-1}|\delta t - (1 - s_L)\delta y)^2(1/\tan(\beta))$ ;
32     else
33        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x - (1 - c_{i,j})\delta x\delta y$ ;
34     end
35   end

```

Algorithm 42 : Flux Calculator, y-sweep, C-Cells, Part 2

```

24 else if Case III then
25   if  $v_{i,j} > 0$  then
26      $F_{t,i,j} = v_{i,j} \delta t (s_T \delta x - 0.5 * v_{i,j} \delta t (1 / \tan(\beta)))$ ;
27   end
28   if  $v_{i,j-1} < 0$  then
29      $F_{b,i,j} = |v_{i,j-1}| \delta t (s_B \delta x + 0.5 |v_{i,j-1}| \delta t (1 / \tan(\beta)))$ ;
30   end
31 else
32   if  $v_{i,j} > 0$  then
33     if  $v_{i,j} \delta t \leq s_R \delta y$  then
34        $F_{t,i,j} = v_{i,j} \delta t \delta x$ ;
35     else
36        $F_{t,i,j} = v_{i,j} \delta t \delta x - 0.5 (v_{i,j} \delta t - s_R \delta y)^2 (1 / \tan(\beta))$ ;
37     end
38   end
39   if  $v_{i-1,j} < 0$  then
40     if  $v_{i,j-1} \delta t \geq (1 - s_R) \delta x$  then
41        $F_{b,i,j} = |v_{i,j-1}| \delta t \delta x - (1 - c_{i,j}) \delta x \delta y$ ;
42     else
43        $F_{b,i,j} = |v_{i,j-1}| \delta t (s_B \delta x + 0.5 |v_{i,j-1}| \delta t (1 / \tan(\beta)))$ ;
44     end
45   end
46 end

```

Algorithm 43 : Flux Calculator, x-sweep, D-Cells, Part 1

Input : Volume fraction field, c , Horizontal component velocity field, u , Edge intersections, s_T , s_R , s_B and s_L

Output : Outward volume flux values for D type surface cells, Fl and Fr

```

1 Determine case as in algorithm 13;
2 if Case I then
3   if  $u_{i,j} > 0$  then
4     if  $u_{i,j} \geq s_T \delta x$  then
5        $Fr_{i,j} = c_{i,j} \delta x \delta y$ ;
6     else
7        $Fr_{i,j} = 0.5 u_{i,j} \delta t (2 - u_{i,j} \delta t / (s_T \delta x)) s_R \delta y$ ;
8     end
9   end
10  if  $u_{i-1,j} < 0$  then
11    if  $|u_{i-1,j}| \leq (1 - s_T) \delta x$  then
12       $Fl_{i,j} = 0$ ;
13    else
14       $Fl_{i,j} = 0.5 (|u_{i-1,j}| \delta t - (1.0 - s_T) \delta x)^2 \tan(\beta)$ ;
15    end
16  end
17 else if Case II then
18   Use Case III equations if  $u_{i,j} > 0$  then
19     if  $u_{i,j} \delta t \leq s_B \delta x$  then
20        $Fr_{i,j} = u_{i,j} \delta t \delta y$ ;
21     else if  $u_{i,j} \delta t \leq s_T \delta x$  then
22        $Fr_{i,j} = u_{i,j} \delta t \delta y - 0.5 (u_{i,j} * \delta t - s_B \delta x)^2 \tan(\beta)$ ;
23     else
24        $Fr_{i,j} = c_{i,j} \delta x \delta y$ ;
25     end
26   end
27   if  $u_{i-1,j} < 0$  then
28     if  $|u_{i-1,j}| \delta t \leq (1 - s_T) \delta x$  then
29        $Fl_{i,j} = 0$ ;
30     else if  $|u_{i-1,j}| \delta t \leq (1 - s_B) \delta x$  then
31        $Fl_{i,j} = 0.5 (|u_{i-1,j}| \delta t - (1.0 - s_T) \delta x)^2 \tan(\beta)$ ;
32     else
33        $Fl_{i,j} = |u_{i-1,j}| \delta t \delta y - (1 - c_{i,j}) \delta x \delta y$ ;
34     end
35   end

```

Algorithm 44 : Flux Calculator, x-sweep, D-Cells, Part 2

```

36 else if Case III then
37   Use Case II equations if  $u_{i,j} > 0$  then
38      $Fr_{i,j} = u_{i,j} \delta t (s_R \delta y - 0.5 * u_{i,j} \delta t \tan(\beta));$ 
39   end
40   if  $u_{i-1,j} < 0$  then
41      $Fl_{i,j} = |u_{i-1,j}| \delta t (s_L \delta y + 0.5 |u_{i-1,j}| \delta t \tan(\beta));$ 
42   end
43 else
44   if  $u_{i,j} > 0$  then
45     if  $u_{i,j} \leq s_B \delta x$  then
46        $Fr_{i,j} = u_{i,j} \delta t \delta y;$ 
47     else
48        $Fr_{i,j} = u_{i,j} \delta t \delta y - 0.5 \tan(\beta) (u_{i,j} \delta t - s_B \delta x)^2;$ 
49     end
50   end
51   if  $u_{i-1,j} < 0$  then
52     if  $|u_{i-1,j}| \geq (1 - s_B) \delta x$  then
53        $Fl_{i,j} = |u_{i-1,j}| \delta t \delta y - (1 - c_{i,j}) \delta x \delta y;$ 
54     else
55        $Fl_{i,j} = |u_{i-1,j}| \delta t (s_L \delta y + 0.5 |u_{i-1,j}| \delta t \tan(\beta));$ 
56     end
57   end
58 end

```

Algorithm 45 : Flux Calculator, y-sweep, D-Cells, Part 1

Input : Volume fraction field, c , Horizontal component velocity field, v , Edge intersections, s_T , s_R , s_B and s_L

Output : Outward volume flux values for D type surface cells, F_t and F_b

```

1 Determine case as in algorithm 13;
2 if Case I then
3   if  $v_{i,j} > 0$  then
4     if  $v_{i,j}\delta t \geq s_R\delta x$  then
5        $F_{t,i,j} = c_{i,j}\delta x\delta y$ ;
6     else
7        $F_{t,i,j} = 0.5v_{i,j}\delta t(2 - v_{i,j}\delta t/(s_R\delta y))s_T\delta x$ ;
8     end
9   end
10  if  $v_{i,j-1} < 0$  then
11    if  $|v_{i,j-1}|\delta t \leq (1 - s_R)\delta y$  then
12       $F_{b,i,j} = 0$ ;
13    else
14       $F_{b,i,j} = 0.5(|v_{i,j-1}|\delta t - (1 - s_R)\delta y)^2(1/\tan(\beta))$ ;
15    end
16  end
17 else if Case II then
18   Use Case III equations if  $v_{i,j} > 0$  then
19      $F_{t,i,j} = v_{i,j}\delta t(s_T\delta x - 0.5 * v_{i,j}\delta t(1/\tan(\beta)))$ ;
20   end
21   if  $v_{i,j-1} < 0$  then
22      $F_{b,i,j} = |v_{i,j-1}|\delta t(s_B\delta x + 0.5|v_{i,j-1}|\delta t(1/\tan(\beta)))$ ;
23   end

```

B.3 SEDIMENT SOLUTION

The algorithms presented below represent the procedure for the calculation of the change in bed heights and the change in suspended sediment in the system.

Algorithm 46 : Flux Calculator, y-sweep, D-Cells, Part 2

```

36 else if Case III then
37   Use Case II equations if  $v_{i,j} > 0$  then
38     if  $v_{i,j}\delta t \leq s_L\delta y$  then
39        $F_{t,i,j} = v_{i,j}\delta t\delta x$ ;
40     else if  $u_{i,j}\delta t \leq s_T\delta x$  then
41        $F_{t,i,j} = v_{i,j}\delta t\delta x - 0.5(v_{i,j} * \delta t - s_L\delta y)^2(1/\tan(\beta))$ ;
42     else
43        $F_{t,i,j} = c_{i,j}\delta x\delta y$ ;
44     end
45   end
46   if  $v_{i,j-1} < 0$  then
47     if  $|v_{i,j-1}|\delta t \leq (1 - s_R)\delta y$  then
48        $F_{b,i,j} = 0$ ;
49     else if  $|v_{i,j-1}|\delta t \leq (1 - s_L)\delta y$  then
50        $F_{b,i,j} = 0.5(|v_{i,j-1}|\delta t - (1 - s_R)\delta y)^2(1/\tan(\beta))$ ;
51     else
52        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x - (1 - c_{i,j})\delta x\delta y$ ;
53     end
54   end
55 else
56   if  $v_{i,j} > 0$  then
57     if  $v_{i,j}\delta t \leq s_L\delta y$  then
58        $F_{t,i,j} = v_{i,j}\delta t\delta x$ ;
59     else
60        $F_{t,i,j} = v_{i,j}\delta t\delta x - 0.5(v_{i,j}\delta t - s_L\delta y)^2(1/\tan(\beta))$ ;
61     end
62   end
63   if  $v_{i,j-1} < 0$  then
64     if  $|v_{i,j-1}|\delta t \geq (1 - s_L)\delta y$  then
65        $F_{b,i,j} = |v_{i,j-1}|\delta t\delta x - (1 - c_{i,j})\delta x\delta y$ ;
66     else
67       Set outward flux from cell (i,j)s bottom edge,  $F_{b,i,j}$ , to
         $|v_{i,j-1}|\delta t(s_B\delta x + 0.5|v_{i,j-1}|\delta t(1/\tan(\beta)))$ ;
68     end
69   end
70 end

```

Algorithm 47 : Peak, Trough, Jump and Plateau Finder

Input : A vector of bed heights, h **Output** : A vector of Peak/Trough/Jump and Plateau locations,
PTL

```

1  for  $i \leftarrow 2$  to  $\text{ncols} - 1$  do
2      if  $h[i]$  is equal to  $h[i-1]$  then
3          if PreviousPlateau is false then
4              Set PlateauStart to  $(i - 1)$ ;
5              Set PreviousPlateau to true;
6          end
7      else if (PreviousPositive is true or PreviousPlateau is true) and
           $h[i] - h[i - 1] > \delta y$  then
8          Jump: Save index  $i$  and  $(i - 1)$  in PTL;
9      else if (PreviousPositive is false or PreviousPlateau is true) and
           $h[i - 1] - h[i] > \delta y$  then
10         Jump: Save index  $i$  and  $(i - 1)$  in PTL;
11     else if PPl is true then
12         Plateau: Save index  $\left(\frac{(i-1)-\text{PlateauStart}}{2} + \text{PlateauStart}\right)$ 
13         in PTL
14         Set PreviousPlateau to false;
15     else if PPo is true and  $h[i] < h[i - 1]$  then
16         Peak: Save index  $i$  in PTL;
17         Set PreviousPositive to false;
18     else if PPo is false and  $h[i] > h[i - 1]$  then
19         Trough: Save index  $i$  in PTL;
20         Set PreviousPositive to true;
21     end
22 end

```

Algorithm 48 : Angle of repose height redistribution

Input : A vector of bed heights, h and a vector of Peak/Trough/Jump and Plateau locations, PTL

Output : A vector redistributed bed heights, h

```

1  for  $a \leftarrow 1$  to Size of PTL do
2      if  $h[PTL[a]] \geq h[PTL[a - 1]]$  then
3          if Gradient > CriticalGradient then
4              for  $b \leftarrow PTL[a]$  to  $PTL[a - 1]$ ,  $b --$  do
5                  Add the height  $h[b]$  minus the trough height
                      $h[PTL[a] - 1]$  to a sum
6              end
7              Calculate Triangle Base and height
8              Calculate the heights at all cell centre points within the
                     triangle
9          end
10     else
11         if Gradient > CriticalGradient then
12             for  $b \leftarrow PTL[a - 1]$  to  $PTL[a]$ ,  $b ++$  do
13                 Add the height  $h[b]$  minus the trough height
                      $h[PTL[a] - 1]$  to a sum
14             end
15             Calculate Triangle Base and height
16             Calculate the heights at all cell centre points within the
                     triangle
17         end
18     end
19 end

```

Algorithm 49 : Sediment Procedure, part 1

Input : Volume fraction field, c , Sediment concentration field C ,
bed height vector h , component velocity fields u and v ,
sediment particle diameter d , particle density ρ_d , critical
angle of repose

Output : New values for Volume fraction field, c , Sediment
concentration field C , bed height vector h

```

1 Sum mass of sediment in system
2 Calculate Critical shear stress, Shields stress
3 Calculate flow depth  $d_f$  for each width
4 Calculate Diffusion Coefficients
5 if SweepDirection is false then
6     Find Free Surface and Calculate horizontal volume fluxes
7     Apply Volume Flux boundary conditions
8     if Control Volume is not full then
9         Advect fluid volumes horizontally and apply boundary
          conditions
10    end
11    Advect and diffuse sediment concentrations horizontally
12    Find Free Surface and Calculate vertical volume fluxes from
        volume fraction copy with over/undershoots removed
13    Apply Volume Flux boundary conditions
14    if Control Volume is not full then
15        Advect fluid volumes vertically and apply boundary
          conditions
16    end
17    Advect, diffuse and settle sediment concentrations vertically
18    Calculate Bed Transport and changes to concentration due to
        deposition, erosion and bed load
19    Angle of Repose Redistribution: Find PTL (see algorithm 47),
        and Calculate new values for  $h$  (see algorithm 48)
20    Move Left overs up
21    Set Flags
22    Set SweepDirection to true

```

Algorithm 50 : Sediment Procedure, part 2

```

23 else
24   Find Free Surface and Calculate vertical volume fluxes
25   if Control Volume is not full then
26     | Advect fluid volumes vertically
27   end
28   Advect, diffuse and settle sediment concentrations vertically
29   Find Free Surface and Calculate horizontal volume fluxes
30   if Control Volume is not full then
31     | Advect fluid volumes horizontally
32   end
33   Advect and diffuse sediment concentrations horizontally
34   Calculate Bed Transport and changes to concentration due to
      deposition, erosion and bed load
35   Angle of Repose Redistribution: Find PTL (see algorithm 47),
      and Calculate new values for h (see algorithm 48)
36   Move Left overs up
37   Set Flags
38   Set SweepDirection to false
39 end
40 Remove overshoots and undershoots, for volume fraction and
      sediment concentration
41 Impose variable bed height velocity boundary conditions
42 Check Sediment Conservation of Mass

```

BIBLIOGRAPHY

- A.D. Abrahams. Bed-load transport equation for sheet flow. *Journal of Hydraulic Engineering*, 129(2):159–163, 2003.
- R. J. Adrian and J. Westerweel. *Particle image velocimetry*. Number 30. Cambridge University Press, 2011.
- A.A. Amsden and F.H. Harlow. The smac method: a numerical technique for calculating incompressible fluid flows. *Los Alamos Sci. Lab. Report LA-4370*, 1970.
- Association of state dam safety officials ASDSO. List of dam and levee failures (dam failures, dam incidents (near failures)), 2013. URL http://www.damsafety.org/media/Documents/PDF/US_FailuresIncidents.pdf.
- R.A. Bagnold. An approach to the sediment transport problem from general physics: Us geol. *Survey Prof. Paper*, 422(1):37, 1966.
- R.A. Bagnold and G. Taylor. Motion of waves in shallow water. interaction between waves and sand bottoms. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 187(1008):1–18, 1946.
- R.A. Bagnold and G. Taylor. Mechanics of marine sedimentation. *The Sea v. 3: The Earth Beneath the Sea*, pages 507–582, 1963.
- S.W. Bell, R.C. Elliot, and M. Hanif Chaudhry. Experimental results of two-dimensional dam-break flows. *Journal of Hydraulic Research*, 30(2):225–252, 1992.
- G. Bellaiche. Etude géodynamique de la marge continentale au large du massif des maures (var) et de la plaine abyssale ligure, 1969.
- R. Bernetti, V.A. Titarev, and E.F. Toro. Exact solution of the riemann problem for the shallow water equations with discontinuous bottom geometry. *Journal of Computational Physics*, 227(6):3212–3243, 2008.
- W.L. Briggs, V.E. Henson, and S.F. McCormick. *A multigrid tutorial*, volume 72. Siam, 2000.
- M. Burkow and M. Griebel. Numerical simulation of sediment processes and resulting bedforms-application to scouring at an obstacle. 2013.

- B. Camenen and M. Larson. A general formula for non-cohesive bed load sediment transport. *Estuarine, Coastal and Shelf Science*, 63(1): 249–260, 2005.
- Z. Cao, G. Pender, S. Wallis, and P. Carling. Computational dam-break hydraulics over erodible sediment bed. *Journal of hydraulic engineering*, 130(7):689–703, 2004.
- H. Capart. Dam-break induced geomorphic flows, 2000.
- H. Capart and DL Young. Formation of a jump by the dam-break wave over a granular bed. *Journal of Fluid Mechanics*, 372:165–187, 1998.
- T. Capone. Sph numerical modelling of impulse water waves generated by landslides, 2009.
- M.A. Carrigy. Experiments on the angles of repose of granular materials¹. *Sedimentology*, 14(3-4):147–158, 1970.
- W. L. Chadwick and A. Casagrande. Report to us department of the interior and state of idaho on failure of teton dam. Independent panel to review cause of Teton dam failure, 1976.
- N.S. Cheng. Exponential formula for bedload transport. *Journal of Hydraulic Engineering*, 128(10):942–946, 2002.
- A. J. Chorin. Numerical solution of the navier-stokes equations. *Mathematics of computation*, 22(104):745–762, 1968.
- A.J. Chorin. Flame advection and propagation algorithms* 1. *Journal of Computational Physics*, 35(1):1–11, 1980.
- S.J. Cummins, M.M. Francois, and D.B. Kothe. Estimating curvature from volume fractions. *Computers & structures*, 83(6):425–434, 2005.
- S.B. Dalziel. Rayleigh-taylor instability: experiments with image analysis. *Dyn. Atmos. Ocean*, 20:127–153, 1993.
- P.G. De Gennes. Wetting: statics and dynamics. *Reviews of modern physics*, 57(3):827, 1985.
- W.E. Dietrich. Settling velocity of natural particles. *Water Resources Research*, 18(6):1615–1626, 1982.
- A. Donev, I. Cisse, D. Sachs, E.A. Variano, F.H. Stillinger, R. Connelly, S. Torquato, and P.M. Chaikin. Improving the density of jammed disordered packings using ellipsoids. *Science*, 303(5660):990–993, 2004.
- R.F. Dressler. Hydraulic resistance effect upon the dam-break functions. *Journal of Research of the National Bureau of Standards*, 49(3): 217–225, 1952.

- R.F. Dressler. Comparison of theories and experiments for the hydraulic dam-break wave. In *Proc. Intl Assoc. of Scientific Hydrology Assemblée Générale*, pages 319–328, 1954.
- I.B. Eguiazaroff. Regulation of the water level in the reaches of canalized rivers. In *XVI International Congress of Navigation, Brussels*, 1935.
- H.A. Einstein. *The bed-load function for sediment transportation in open channel flows*. Number 1026. US Dept. of Agriculture, 1950.
- K. El Kadi Abderrezzak and A. Paquier. Applicability of sediment transport capacity formulas to dam-break flows over movable beds. *Journal of Hydraulic Engineering*, 137(2):209–221, 2010.
- J.W. Elder. The dispersion of marked fluid in turbulent shear flow. *Journal of fluid mechanics*, 5(04):544–560, 1959.
- F.M. Exner. Über die wechselwirkung zwischen wasser und geschiebe in flüssen. *Sitzungsber. Akad. Wiss. Wien Abt. Ila*, 134:166–204, 1925.
- H. Fahlbusch. Early dams. *Proceedings of the ICE-Engineering History and Heritage*, 162(1):13–18, 2009.
- V.G. Ferreira, M.F. Tomé, N. Mangiavacchi, A. Castelo, J.A. Cuminato, A.O. Fortuna, and S. McKee. High-order upwinding and the hydraulic jump. *International journal for numerical methods in fluids*, 39(7):549–583, 2002.
- L. Fraccarollo and H. Capart. Riemann wave description of erosional dam-break flows. *Journal of Fluid Mechanics*, 461:183–228, 2002.
- R.A. Gentry, R.E. Martin, and B.J. Daly. An Eulerian differencing method for unsteady compressible flow problems. *Journal of Computational Physics*, 1(1):87–118, 1966.
- U. Ghia, K. N. Ghia, and C. T. Shin. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of Computational Physics*, 48(3):387–411, 1982.
- R.A. Gingold and J.J. Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*, 181(3):375–389, 1977.
- W.J. Graham. A procedure for estimating loss of life caused by dam failure, 1999.
- M. Griebel, T. Dornseifer, and T. Neunhoeffler. *Numerical simulation in fluid dynamics: a practical introduction*. Society for Industrial Mathematics, 1998.

- T.C. Hales. A proof of the kepler conjecture. *Annals of mathematics*, pages 1065–1185, 2005.
- T.C. Hales, M. Adams, G. Bauer, D.T. Dang, J. Harrison, T.L. Hoang, C. Kaliszyk, V. Magron, S. McLaughlin, T.T. Nguyen, et al. A formal proof of the kepler conjecture. *arXiv preprint arXiv:1501.02155*, 2015.
- A. Harten, P.D. Lax, and B.v. Leer. On upstream differencing and godunov-type schemes for hyperbolic conservation laws. *SIAM review*, 25(1):35–61, 1983.
- M.R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):409–436, 1952.
- C.W. Hirt and B.D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries* 1. *Journal of computational physics*, 39(1):201–225, 1981.
- C.W. Hirt and J.P. Shannon. Free-surface stress conditions for incompressible-flow calculations* 1. *Journal of Computational Physics*, 2(4):403–411, 1968.
- C.W. Hirt, B.D. Nichols, and N.C. Romero. SOLA: A numerical solution algorithm for transient fluid flows. *NASA STI/Recon Technical Report N*, 75:32418, 1975.
- F. Hjulstrom. *Studies of the morphological activity of rivers as illustrated by the River Fyris: Inaugural Dissertation*, volume 25. Almqvist & Wiksells, 1935.
- F. Hjulstrom. Transportation of detritus by moving water: Part 1. transportation. *Recent marine Sediments, Amer. Assoc. Petrol. Geol.*, pages 5–31, 1939.
- A.J. Hogg and D. Pritchard. The effects of hydraulic resistance on dam-break and other shallow inertial flows. *Journal of Fluid Mechanics*, 501(-1):179–212, 2004.
- J.M. Hyman. Numerical methods for tracking interfaces. *Physica D: Nonlinear Phenomena*, 12(1-3):396–407, 1984.
- H.M. Jaeger, S.R. Nagel, et al. Physics of the granular state. *Science*, 255(5051):1523–1531, 1992.
- S.K. Jain, P.K. Agarwal, and V.P. Singh. *Hydrology and water resources of India*. Springer, 2007.
- D.M. Kelly and N. Dodd. Floating grid characteristics method for unsteady flow over a mobile bed. *Computers & Fluids*, 38(4):899–909, 2009.

- J. Kepler. Strena seu de nive sexangula (the six-cornered snowflake). 1611.
- P.M.J. Kerssens, L.C. van Rijn, and A. Prins. Model for suspended sediment transport. *Journal of the Hydraulics Division*, 105(5):461–476, 1979.
- G. Kesserwani, A. Shamkhalchian, and M.J. Zadeh. Fully coupled discontinuous galerkin modeling of dam-break flows over movable bed with sediment transport. *Journal of Hydraulic Engineering*, 140(4), 2014.
- A.C. King and D.J. Needham. The initial development of a jet caused by fluid, body and free-surface interaction. part 1. a uniformly accelerating plate. *Journal of Fluid Mechanics*, 268(-1):89–101, 1994.
- P.D. Komar and M.C. Miller. The threshold of sediment movement under oscillatory water waves. *Journal of Sedimentary Research*, 43(4), 1973.
- A. Korobkin and O. Yilmaz. The initial stage of dam-break flow. *Journal of Engineering Mathematics*, 63(2):293–308, 2009.
- D.B. Kothe and W.J. Rider. A comparison of interface tracking methods. Technical report, Los Alamos National Lab., NM (United States), 1995a.
- D.B. Kothe and W.J. Rider. Comments on modeling interfacial flows with Volume-of-Fluid methods. *Los Alamos National Laboratory. LA-UR-94-3384*, 1995b.
- D.B. Kothe, R.C. Mjolsness, and M.D. Torrey. Ripple: A computer program for incompressible flows with free surfaces. *Los Alamos National Laboratory Rep.*, 1991.
- G. Lauber and W.H. Hager. Experiments to dambreak wave: Horizontal channel. *Journal of hydraulic research*, 36(3):291–307, 1998.
- J. Leal, R.M.L. Ferreira, A.B. Franco, and A.H. Cardoso. Dam-Break Waves Over Movable Bed Channels Experimental Study. In *Proceedings of the Congress-International Association for Hydraulic Research*, pages 232–239, 2001.
- G.R. Lesser, J.A. Roelvink, J.A.T.M. Van Kester, and G.S. Stelling. Development and validation of a three-dimensional morphological model. *Coastal engineering*, 51(8):883–915, 2004.
- L. Levin. Mouvement non permanent sur les cours d’eau à la suite de rupture de barrage. *Rev. Gén. Hydr*, 72, 1952.
- L.B. Lucy. A numerical approach to the testing of the fission hypothesis. *The astronomical journal*, 82:1013–1024, 1977.

- M. Manohar. Mechanics of bottom sediment movement due to wave action. Technical report, U.S. Army Corps of Engineers Beach Erosion Board, Tech. memo. No. 47, 48p., 1955.
- R. Marsooli and W. Wu. Three-dimensional numerical modeling of dam-break flows with sediment transport over movable beds. *Journal of Hydraulic Engineering*, 141(1), 2014.
- K. Marx. Letter to Joseph Weydemeyer, June 1851.
- N. McMullin. Early stage dam break: An investigation into experimental apparatus and procedure. Master's thesis, University of Nottingham, 2010.
- E. Meyer-Peter and R. Müller. Formulas for bed-load transport. In *Proceedings of the 2nd Meeting of the International Association for Hydraulic Structures Research*, pages 39–64, 1948.
- J.J. Monaghan. Simulating free surface flows with sph. *Journal of computational physics*, 110(2):399–406, 1994.
- T. Mulder, S. Zaragosi, J.M. Jouanneau, G. Bellaiche, S. Guérinaud, and J. Querneau. Deposits related to the failure of the malpasset dam in 1959: an analogue for hyperpycnal deposits from jökulhlaups. *Marine Geology*, 260(1):81–89, 2009.
- R.J. Munro, N. Bethke, and S.B. Dalziel. Sediment resuspension and erosion by vortex rings. *Physics of Fluids (1994-present)*, 21(4):046601, 2009.
- B.D. Nichols, C.W. Hirt, and R.S. Hotchkiss. Sola-vof: A solution algorithm for transient fluid flow with multiple free boundaries. *Los Alamos Scientific Lab., NM (USA)*, 1980.
- W. Noh and Paul Woodward. Slic (simple line interface calculation). In Adriaan van de Vooren and Pieter Zandbergen, editors, *Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics June 28 to July 2, 1976 Twente University, Enschede*, volume 59 of *Lecture Notes in Physics*, pages 330–340. Springer Berlin / Heidelberg, 1976.
- F.V. Pohle. *The Lagrangian equations of hydrodynamics; solutions which are analytic functions of the time*. PhD thesis, New York University., 1950.
- A.K. Prasad. Stereoscopic particle image velocimetry. *Experiments in fluids*, 29(2):103–116, 2000.
- W.H. Press, Teukolsky, S.A., Vetterling, and B.P. W.T. and Flannery. *Numerical recipes: The Art of Scientific Computing*. Cambridge University Press, 2007.

- S.L. Razavitoosi, S.A. Ayyoubzadeh, and A. Valizadeh. Two-phase sph modelling of waves caused by dam break over a movable bed. *International Journal of Sediment Research*, 29(3):344–356, 2014.
- W.J. Rider and D.B. Kothe. Reconstructing volume tracking. *Journal of computational physics*, 141(2):112–152, 1998.
- A. Ritter. Die fortpflanzung der wasserwellen (the propagation of water waves). *Vereine Deutscher Ingenieure Zeitschrift*, 36(2):33, 1892.
- P.L. Roe. Approximate riemann solvers, parameter vectors, and difference schemes. *Journal of computational physics*, 43(2):357–372, 1981.
- M. Rudman. Volume-tracking methods for interfacial flow calculations. *International journal for numerical methods in fluids*, 24(7):671–691, 1997.
- Y. Saad. *Iterative methods for sparse linear systems*. Siam, 2003.
- A.J.C.B. de Saint Venant. Théorie du mouvement non permanent des eaux. *Comptes-Rendus de l'Académie des Sciences*, 73:147–237, 1871.
- R. Scardovelli and S. Zaleski. Interface reconstruction with least-square fit and split eulerian–lagrangian advection. *International Journal for Numerical Methods in Fluids*, 41(3):251–274, 2003.
- A. Schoklitsch. Über dambruchwellen.". *Sitzungsberichten der Königliche Akademie der Wissenschaften, Vienna*, 126:1489–1514, 1917a.
- A. Schoklitsch. Ueber dambruchellen. *Sitzber Akad. Wiss. Wein.*, 126, 1917b.
- P. Shaffner. Turning geologic data into knowledge. *AEG Dam Symposium*, 2011. URL http://www.aegweb.org/docs/events/ds11_summary_of_dam_failure_case_histories.pdf.
- S. Shao and E.Y.M. Lo. Incompressible sph method for simulating newtonian and non-newtonian flows with a free surface. *Advances in Water Resources*, 26(7):787–800, 2003.
- A. Shields. Application of similarity principles and turbulence research to bed-load movement. 1936. Anwendung der aenlichkeitsmechanik und der turbulenzforschung auf die geschiebebewegung. Mitteilungen der Preussischen Versuchsanstalt fur Wasserbau und Schiffbau, Berlin, Germany, translated to English by W. P. Ott and J. C. van Uchelen, CalTech, Pasadena, CA.
- A.B. Shvidchenko and G. Pender. Flume study of the effect of relative depth on the incipient motion of coarse uniform sediments. *Water Resources Research*, 36(2):619–628, 2000.

- I.J. Smalley and T.A. Dijkstra. The teton dam (idaho, usa) failure: problems with the use of loess material in earth dam structures. *Engineering geology*, 31(2):197–203, 1991.
- G.M. Smart and M.N.R Jaggi. Sediment transport on steep slopes, mitteilungen der versuchsanstalt fur wasserbau. *Hydrologie und Glaziologie*, 64.
- S. Soares-Frazão, R. Canelas, Z. Cao, L. Cea, H.M. Chaudhry, A. Die Moran, K. El Kadi, R. Ferreira, I.F. Cadórniga, N. Gonzalez-Ramirez, et al. Dam-break flows over mobile beds: experiments and benchmark tests for numerical models. *Journal of Hydraulic Research*, 50(4):364–375, 2012.
- M. Soltanpour and S.M.H. Jazayeri. Numerical modeling of suspended cohesive sediment transport and mud profile deformation. In *10th International Coastal Symposium*, volume Special Issue 56, pages 663–667. Journal of Coastal Research, 2009.
- B. Spinewine. Two-layer shallow water modelling of fast geomorphic flows and experimental validation on idealized laboratory dam-break waves. *31st IAHR Congress Seoul*, 2005.
- B. Spinewine and Y. Zech. Dam-break waves on a movable bed: a test case exploring different bed materials and an initial bed discontinuity. In *EC Contract EVG1-CT-2001-00037 IMPACT Investigation of Extreme Flood Processes and Uncertainty, Proceedings 3rd Project Workshop, Louvain-la-Neuve, Belgium*, pages 6–7, 2003.
- B. Spinewine and Y. Zech. Small-scale laboratory dam-break waves on movable beds. *Journal of Hydraulic Research*, 45(S1):73–86, 2007.
- P.K. Stansby, A. Chegini, and T.C.D. Barnes. The initial stages of dam-break flow. *J. Fluid Mech.*, 374:407–424, 1998.
- J.J. Stoker. Water waves. the mathematical theory with applications, interscience publ. Inc., New York, 1957.
- G.G. Stokes. *On the effect of the internal friction of fluids on the motion of pendulums*, volume 9. Pitt Press, 1851.
- H.Z. Su, J. Hu, and Z.P. Wen. Optimization of reinforcement strategies for dangerous dams considering time-average system failure probability and benefit–cost ratio using a life quality index. *Natural hazards*, 65(1):799–817, 2013.
- A. Sundborg. The river klarälven: a study of fluvial processes. *Geografiska Annaler*, pages 125–237, 1956.
- M. Sussman. A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. *Journal of Computational Physics*, 187(1):110–136, 2003.

- C. Swartenbroekx, S. Soares-Frazão, B. Spinewine, V. Guinot, and Y. Zech. Hyperbolicity preserving hll solver for two-layer shallow-water equations applied to dam-break flows. *Proceedings of River Flow, Dittrich A, Koll K, Aberle J, Geisenhainer P (eds). Bundesanstalt für Wasserbau: Braunschweig*, pages 1379–1387, 2010.
- M.F. Tome and S. McKee. GENSMAC: A computational marker and cell method for free surface flows in general domains. *Journal of Computational Physics*, 110(1):171–186, 1994.
- E.F. Toro. *Shock-capturing methods for free-surface shallow flows*. Wiley, 2001.
- E.F. Toro, M. Spruce, and W. Speares. Restoration of the contact surface in the hll-riemann solver. *Shock waves*, 4(1):25–34, 1994.
- Bureau of Reclamation US Department of the Interior. Glen canyon dam beach/habitat-building test flow. draft environmental assessment and draft finding of no significant impact, 1996.
- L.C. Van Rijn. Sediment transport, part i: bed load transport. *Journal of hydraulic engineering*, 110(10):1431–1456, 1984a.
- L.C. Van Rijn. Sediment transport, part ii: Suspended load transport. *Journal of Hydraulic Engineering*, 110(11):1613–1641, 1984b.
- G.B. Whitham. The effects of hydraulic resistance in the dam-break problem. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 227(1170):399, 1955.
- P.R. Wilcock. Estimating local bed shear stress from velocity observations. *Water Resources Research*, 32(11):3361–3366, 1996.
- M. Williams. Numerical methods for tracking interfaces with surface tension in 3d mold filling processes., 2000.
- E. Wohl. *Mountain rivers revisited*, volume 19. John Wiley & Sons, 2013.
- B. Wu, D.S. Van Maren, and L. Li. Predictability of sediment transport in the yellow river using selected transport formulas. *International Journal of Sediment Research*, 23(4):283–298, 2008.
- W. Wu and S.S. Wang. One-dimensional modeling of dam-break flow over movable beds. *Journal of hydraulic engineering*, 133(1):48–58, 2007.
- J. Xia, B. Lin, R.A. Falconer, and G. Wang. Modelling dam-break flows over mobile beds using a 2d coupled approach. *Advances in Water Resources*, 33(2):171–183, 2010.

- D.L. Youngs. Time-dependent multi-material flow with large fluid distortion. *Numerical methods for fluid dynamics*, 1(1):41–51, 1982.
- S.T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics*, 31(3):335–362, 1979.
- Y. Zech, S. Soares-Frazão, B. Spinewine, and N. Le Grelle. Dam-break induced sediment movement: Experimental approaches and numerical modelling. *Journal of Hydraulic research*, 46(2):176–190, 2008.
- M. Zhang, Y. Xu, Z. Hao, and Y. Qiao. Integrating 1d and 2d hydrodynamic, sediment transport model for dam-break flow using finite volume method. *Science China Physics, Mechanics and Astronomy*, 57(4):774–783, 2014.
- M.L. Zhang and W.M. Wu. A two dimensional hydrodynamic and sediment transport model for dam break based on finite volume method with quadtree grid. *Applied Ocean Research*, 33(4):297–308, 2011.
- S. Zhang and J.G. Duan. 1d finite volume model of unsteady flow over mobile bed. *Journal of Hydrology*, 405(1):57–68, 2011.
- F. Zhu. 1d morphodynamical modelling of swash zone beachface evolution, 2012.

“Hurry up please. It’s time”
— William S. Burroughs, *The Western Lands*

COLOPHON

This document was typeset using the typographical look-and-feel classicthesis developed by André Miede. The style was inspired by Robert Bringhurst’s seminal book on typography *“The Elements of Typographic Style”*. classicthesis is available for both L^AT_EX and L_YX:

<http://code.google.com/p/classicthesis/>

Final Version as of May 12, 2015 (classicthesis version 1.0).